



# *OPEN-REALTY*

**2.x**

**DOCUMENTATION**

# TABLE OF CONTENTS

<b>1. GENERAL INFORMATION.....</b>	<b>6</b>
1. What is Open-Realty®?.....	6
2. Open-Realty® Features.....	6
1. Features For Realtors .....	6
2. Features For Designers & Programmers .....	6
3. Open-Realty® Requirements .....	7
4. License Information .....	8
5. Open-Realty® License .....	8
6. Getting Help With Open-Realty® .....	9
<b>2. INSTALLING OPEN-REALTY® .....</b>	<b>10</b>
1. Getting Started .....	10
2. Running the Installer .....	0
3. Default Username/Password .....	11
4. Search Engine Friendly URLs and Optimization .....	11
<b>3. INSTALLING WYSIWYG EDITORS.....</b>	<b>13</b>
<b>4. UPGRADING OPEN-REALTY® .....</b>	<b>14</b>
1. Getting Started .....	14
2. Running the installer.....	0
3. Search Engine Friendly URLs and Optimization .....	15
<b>5. MOVING TO A NEW HOST .....</b>	<b>17</b>
1. Getting Started .....	17
2. Running the installer.....	0
<b>6. OPEN-REALTY® ADMINISTRATION .....</b>	<b>19</b>
1. Configuration .....	20
1. General .....	20
2. Template.....	21
3. SEO (Search Engine Optimization) .....	22
4. Editor / HTML.....	22
5. Numbers .....	23
6. Uploads/Images .....	24
7. Uploads/Files .....	25
8. Search .....	26
9. VTours .....	27
10. Notify.....	27
11. Users .....	28
12. Listings .....	29
13. Maps.....	30
14. Vcards .....	31
15. RSS .....	31
16. Help Links.....	32
17. Blog.....	32
2. Page Editor.....	33
3. Blog Editor .....	34
1. Blog Manager (Navigation View) .....	34
2. Blog Manager (Editor View).....	34
3. Comments .....	35
4. Listing Template Editor.....	36
1. Add or Edit Listing Fields.....	36
2. Set Field Order .....	39
3. Search Setup .....	39
4. Search Results .....	40
5. Property Classes.....	41
1. Insert Property Class.....	41
2. Modify Property Class.....	41

6.	Edit Agent/Member Templates.....	42
1.	Add/Edit Agent & Member Fields .....	42
2.	Set Field Order .....	42
7.	User Manager .....	43
8.	Editing Users - Edit My Account .....	44
1.	User Images .....	44
1.	Displaying User Images .....	44
2.	User Files .....	44
1.	Displaying Files.....	45
9.	Adding Listings.....	46
10.	Editing Listings.....	47
1.	Listing Images.....	47
1.	Displaying Images on Listings.....	48
2.	Virtual Tours (VTours) .....	48
1.	Displaying Virtual Tours (VTours) on Listings .....	48
3.	Listing Files.....	49
1.	Displaying Files on Listings .....	49
11.	Add-on Manager.....	50
12.	Site Log .....	51
<b>7.</b>	<b>TEMPLATE DOCUMENTATION .....</b>	<b>52</b>
1.	INSTALLING NEW TEMPLATES .....	53
1.	Upgrading/Updating Templates.....	53
2.	PERMISSION TEMPLATE TAGS .....	54
3.	ADMIN TEMPLATES .....	56
1.	MAIN ADMIN TEMPLATES.....	57
1.	Main Admin Template Tags.....	57
2.	EDIT LISTINGS TEMPLATE .....	59
1.	Edit Listings Template Tags .....	59
4.	PUBLIC SITE TEMPLATES .....	61
1.	MAIN TEMPLATES .....	62
1.	Main Template Tags .....	63
2.	LISTING DETAIL TEMPLATES .....	66
1.	Listing Detail Template Tags.....	67
3.	SEARCH PAGE TEMPLATES .....	73
1.	Custom Search Pages .....	73
2.	Multiple Sortby Values .....	74
3.	Additional Search String Variables .....	74
4.	Search Page Template Tags.....	75
4.	SEARCH RESULTS TEMPLATES.....	77
1.	Search Results Template Tags .....	78
5.	FEATURED/RANDOM LISTINGS TEMPLATES .....	82
1.	Featured/Random Template Tags .....	83
6.	SEARCH PROPERTY CLASS TEMPLATES .....	85
1.	Search Property Class Template Tags .....	85
7.	VIEW USERS TEMPLATE.....	86
1.	View Users Template Tags.....	87
8.	VIEW AGENT TEMPLATE.....	88
1.	View Agent Template Tags .....	89
9.	VIRTUAL TOUR (VTOUR) TEMPLATES.....	90
1.	VTour Template Tags.....	91
10.	FILE DOWNLOAD TEMPLATES .....	93
1.	File Download Template Tags .....	94
11.	RSS TEMPLATE .....	95
1.	RSS Template Tags .....	96
12.	NEXT/PREV TEMPLATE .....	97
1.	Next/Prev Template Tags .....	98
13.	LISTINGS NOTIFICATION TEMPLATE .....	101

1. Listings Notification Template Tags.....	102
14. TAB PANE CODE DOCUMENTATION .....	103
1. Tabbed Content .....	103
<b>8. AUTOMATING TASKS.....</b>	<b>105</b>
1. Cron jobs .....	105
1. Listings Notification.....	105
<b>9. DEVELOPER'S GUIDE.....</b>	<b>106</b>
1. SVN Access and Usage .....	106
2. Code Standards.....	0
3. Virtual Tour Documentation .....	110
1. Virtual Tours .....	110
2. EGG Solution Virtual Tour Notes: .....	110
3. Creating a Virtual Tour .....	110
4. Uploading and Managing Virtual Tours .....	111
4. Distance Search DB .....	112
<b>10. ADD-ON INFORMATION .....</b>	<b>113</b>
1. INSTALLING AN ADD-ON .....	114
2. ADD-ON DEVELOPER GUIDE .....	115
1. Javascript.....	115
2. Search Results .....	115
3. Add-on Functions.....	116
1. name_install_addon().....	116
2. name_show_admin_icons().....	117
3. name_load_template() .....	117
4. name_run_action_user_template().....	117
5. name_run_action_admin_template() .....	118
6. name_run_template_user_fields().....	118
7. name_addonmanager_help().....	118
8. name_checkupdate_url() .....	119
9. name_update_url(\$version) .....	119
10. name_uninstall_tables() .....	119
4. Add-on Specific Functions .....	120
<b>11. LANGUAGE GUIDE .....</b>	<b>121</b>
1. Custom changes to language variables .....	121
2. Translate Open-Realty® into a different language.....	121
<b>12. FREQUENTLY ASKED QUESTIONS (FAQ's) .....</b>	<b>123</b>
1. I'd like to remove the hidden Open-Realty® license from the source, how do I get permission to do this? .....	123
2. I'd like to give something back to the Open-Realty® project, how can I help? ...	123
3. How do I backup my Open-Realty® site?.....	123
4. I lost my password, how do I recover my password? .....	124
5. I forgot my admin password, and don't remember or have access to the email address for the admin account. How do I reset the password? .....	124
6. What is the difference between agents and members? .....	125
7. I need to change a word, phrase or how something is said in Open-Realty®, how do I change it?.....	125
8. I need a feature that Open-Realty® doesn't have, what can I do? .....	125
<b>13. TROUBLESHOOTING.....</b>	<b>127</b>
1. I deleted, lost, overwrote or otherwise messed up my common.php file, what do I do?.....	127
2. I receive 404 Errors when trying to view Open-Realty® pages.....	127
3. JPG Images are distorted/dicolored after i upload them .....	127
4. I see this error on my search results page, Notice: A non well formed numeric value encountered in include/search.inc.php .....	128
5. Search results don't display any listings but it says there should be search results .....	128
6. Search Page is slow to display .....	128

7. RSS Feed is broken.....	128
<b>14. RECENT CHANGES .....</b>	<b>129</b>

# GENERAL INFORMATION

## What is Open-Realty®?

Open-Realty® is an open source web based real estate listing management application. It is intended to be both easy to setup and use. Written in PHP, Open-Realty® is designed to be a fast and flexible tool for your real estate website. With a large open-source community working with this application, and solid management from [Transparent Technologies, Inc.](#) Open-Realty® is the top choice for use on your website.

Open-Realty® has been the first choice in open source real estate web site solutions since 2003. Open-Realty® provides easy installation, highly customizable templates, virtual tour support, advanced search capabilities, and many other features. Combine this with commercial add-ons such as our [IDX data importer](#), [RETS data importer](#), and [advanced Google map integration](#) and you have all tools needed to have a top real estate website in your market.

Open-Realty® is released free of charge, to help support the continuing development please consider using [Transparent Technologies, Inc.](#) for your web hosting provider.

## Open-Realty® Features

### Features For Realtors

- Easy installation and configuration.
- Ability to add an unlimited number of fields to listings.
- Support multiple property classes in a single install. (Residential, Commercial, etc..)
- Ability to define different fields for each property class.
- You can define which fields are searchable, through our easy to use administrative interface.
- Upload multiple images at once to listings.
- Open-Realty® automatically creates image thumbnail for you.
- Upload Virtual Tours for your listings.
- Visitors to your site can sign up as members, where they can save searches and get email notifications when new properties are added that match their search criteria.
- You can choose to display listing fields to all users, only members who have registered on your site, or to only other agents who have logged into your site
- Open-Realty® automatically places map, school, and neighborhood information links on your site for you.
- Built in contact form provides an easy way for interested parties to contact you, without you having to place your email address on the site for spammers to gather and use.
- WYSIWYG Editor allows you to edit pages such as the contact and about us pages on your site, without any HTML knowledge.
- Built in search engine optimizations help, ensure that your listings appear on the search engines, when buyers are looking.

### Features For Designers & Programmers

- Easy installation and configuration.
- XHTML 1.0 Compliant Design (Strict DTD)

- Template system to allow easy customization of the main site, of listing pages, view agents pages, search results, featured listings, and more.
- WYSIWYG Editor allows you to give agents the ability to customize pages without being able to change your site design.
- PHP code design is object oriented.
- Add-on system allows you to build many new features for clients, without modifying any core files, allowing them to easily upgrade down the road.
- Source code is open-source and actively maintained with a large support community.

## **Open-Realty® Requirements**

### **Apache based web server**

#### **PHP 4.3.0 or higher**

- **PHP GD Libs Support or Imagemagick is required for thumbnail creation, image resizing and Captcha verification on forms and registration.**
- **Magic Quotes disabled**
- **cURL functions are required for full functionality of the Add-on Manager**

#### **MySQL (4.1 or higher)**

## License Information

### Open-Realty® Trademark Usage

Open-Realty® is a registered trademark of Transparent Technologies, Inc. Use of the trademark by sites offering Open-Realty® templates and add-ons is permitted, provided that the following conditions are all met.

1. Display of the trademark is only in its proper form "Open-Realty®"
2. The following statement is displayed on all pages displaying the trademark, "Open-Realty® is a registered trademark of [Transparent Technologies, Inc.](#)"
3. Use of the trademark in no way implies that the templates or add-ons provided are supported or provided by Transparent Technologies, Inc.

All other uses of the trademark are prohibited without express written permission of Transparent Technologies, Inc.

### Transparent Technologies Product Licensing

Open-Realty® is an open source product. The complete source code and documentation is available for [download](#) from our Web site.

The [Open-Realty License](#) for Open-Realty® permits you to use the software at no charge under two conditions.

1. You have to leave the html comment that reads "*<!--Open-Realty® is distributed by Transparent Technologies and is Licensed under the Open-Realty License. See <http://www.open-realty.org/oslicense.html> for more information.-->*" intact.

2. If you use Open-Realty® in an application you redistribute, the complete source code for your application must be available and freely redistributable under reasonable conditions.

### What does "redistribute" mean?

The term "redistribution" in the Open-Realty® public license means your application is distributed to one or more third parties. Giving an application to customers, even in alpha or beta releases, is redistribution. Giving contractors, affiliates, parent organizations or subsidiaries, business partners or support vendors a copy of the application is generally redistribution.

The following are not redistribution:

- Building an application for use internal to your organization, deployed and managed on your company servers.
- Off-site backups or other software archival procedures.

If you have questions about whether your use of Open-Realty® constitutes redistribution, please [contact us](#).

## Open-Realty® License

Copyright (c) 2009

Transparent Technologies. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. If you use Open-Realty® in an application you redistribute, the complete source code for your application must be available for no more than a reasonable reproduction cost.
2. Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
3. Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.



4. Neither the name of Transparent Technologies nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

5. Open-Realty® is designed to produce computer files for distribution, and the original work may insert into such files text ("Attribution Text") which identifies them as having been produced by Open-Realty®, Open-Realty's distributor Transparent Technologies, and the terms of or a reference to this license. Whilst the files themselves are not subject to this license, your grant to use, distribute and modify this work is conditional upon your agreement not to make any modification to the work which might cause such Attribution Text to be removed, or its insertion to be modified in a manner which is likely to make it materially less readable to those using the files in the manner in which they would customarily be used.

THIS SOFTWARE IS PROVIDED BY TRANSPARENT TECHNOLOGIES AND ITS CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL Transparent Technologies OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

---

## Getting Help With Open-Realty®

First and foremost read this documentation manual. It contains instructions and details for getting started with Open-Realty®. If you have a question use the search feature to search the documentation for your answer. Try several different wordings, or simplify your wording to help maximize your results and find the information you need.

If you still need help, visit the Open-Realty® support forums at <http://www.open-realty.org/support>. Use the search feature on the forums to search for your answer. If you can't find an answer by searching the forums then post a new thread in an appropriate topic.

TIPS for posting on the forum for help:

- Be as detailed as possible.
- Always post details of your problem: exact error messages you are receiving, a link to your site and details of how to reproduce the problem, the version of Open-Realty®, templates you are using etc all help us to give you an answer to your problem.
- Be polite and courteous. NOBODY on the forums is getting paid for their time to help you solve your problem.
- DO NOT DOUBLE POST. Double posting is posting the same question in more than one topic on the forums. It is frustrating to read a post, answer a question and find the same post elsewhere, or find that somebody else has already answered the other post.
- If you post questions pertaining to custom work, coding or modifications do not be surprised or upset if you do not get an answer right away or at all. The people who provide help and answer questions on the forums are not always inclined to do custom coding for people for free.
- Many questions that are posted on the forums ARE in the documentation or already answered on the forums. Make sure you have searched for your question before you post. Most of the people that provide help on the forums will have to look up the answer to your question in the forums or documentation which is what you should be doing in the first place.

# INSTALLING OPEN-REALTY®

## Getting Started

1. [Download](#) the latest official Open-Realty® release.
  1. Go to <http://www.open-realty.org/download.html> to download Open-Realty®.
  2. You can also use SVN to get the latest work in progress. SVN Revisions are not official releases and may have bugs in them. Read the SVN Information in the Developer's Guide section of this documentation.
2. Extract the downloaded file(s) and upload to your website, either to the root folder on your site, usually called www or public html, or to a sub-folder of your root.
3. Set Permissions (chmod) on the following files/folders so that they are writable by the web server. You can set permissions(chmod) using most FTP Clients, Web based File managers, or via a shell account. All of the following files/folders should have permissions of 777 or 755 depending on your host's configuration.
  1. `include/common.dist.php`
  2. `images/listing_photos`
  3. `images/user_photos`
  4. `images/vtour_photos`
  5. `images/page_upload`
4. Create a blank MySQL or PostGRES database on your server. Most hosting providers provide a control panel interface for doing this, you should check with your hosting provider if you are not sure how to do this. You should also create a database user and assign them All Privileges to the database you created. Write down the database name, user-name, and password you just created.
5. Using your web browser, go to `http://www.yourdomain.com/installpath/install/index.php`. For example if you installed this on a domain called `myrealestatecompany.com` in a sub-directory called "open-realty" you would go to, `http://www.myrealestatecompany.com/open-realty/install/index.php`.

## Running the Installer

1. Read the license agreement and click the "Agree" button to continue.
2. Select the default language for your site and click submit. **NOTE:** English is the only language file that is regularly maintained by the Development team. If you have difficulty installing due to blank pages in another language it is likely due to an incomplete or missing lang file for your selected lang. To fix that you can open the /install/lang/en/ folder and replace the lang.inc.php file from your configured lang folder with the one from the 'en' folder.
3. The next screen verifies the permissions your set in step 4. If it finds any permission problems correct them and refresh the page. When all permissions are correct; Click "Click to Continue Installation".
4. You now need to select the type of installation you are doing. The options are as follows.
  1. New Install Of Open-Realty 2.x
  2. Upgrade from Open-Realty 2.x (Beta 1 or higher) (**See the Upgrade guide for details**)
  3. Upgrade from Open-Realty 1.1.5 (**See the Upgrade guide for details**)
  4. Update Path and URL information only - This option should be used when you have moved your Open-Realty install to a new directory.
5. Select the type of database you created in step 4 of "Getting Started" above. Insert your database information from step 5 into the appropriate fields. The last four settings should not be changed unless you know what you are doing. Click next.
6. This step creates the common.php file, which stores the database configuration information for Open-Realty® to use. Click "Continue to setup the database."
7. This step will create and populate the Open-Realty® database tables.
8. Click "Click here to configure your installation". Next you will need to, at the minimum change the Admin email address in the configuration before the site will operate. To log into the admin area and start using Open-Realty® use the login information below. This is the default admin user, you should change the password once logged in.

## Default Username/Password

**Username:** *admin*

**Password:** *password*

## Search Engine Friendly URLs and Optimization

Open-Realty® has the ability to be configured to use Search Engine Friendly (SEF) URLs. These are URLs that appear in the user's browser, and search engines, as a standard html file name instead of the often cryptic URLs in PHP scripts that include variables, actions and values.

To enable SEF URLs there are two steps that are required as follows:

1. Change the name of the included ".htaccess-or" to ".htaccess" on your server. This file contains the rules that are used by the Apache rewrite engine.
2. In the Site Configuration on the **SEO** Tab edit the setting for the **URL Type** option and set it to "**Search Engine Friendly**".

Please read the **Administrators Guide** for more information on how to further configure and customize Open-Realty®.

Congratulations your Open-Realty® installation is complete! You must delete the install directory before your site will be accessible.

# INSTALLING WYSIWYG EDITORS

1. [Download](#) the latest official release of the WYSIWYG Editor(s) you want to install from the Open-Realty® download page. You can also use SVN to get the latest work in progress. SVN Revisions are not official releases and may have bugs in them.
  1. Go to <http://www.open-realty.org/download.html> to download WYSIWYG Editors.
  2. NOTE: If you downloaded Open-Realty® using SVN then the editors are already included and you can skip to step 4.
2. Extract the downloaded WYSIWYG editor to your hard drive. You should have a folder with the name of the editor for example: "fckeditor".
3. Copy the WYSIWYG editor folder to your Open-Realty® installation. You need to place the editor's folder in `/include/class/` in your Open-Realty® installation.
  1. Completed it should look like `/include/class/fckeditor` with the editor's files under this folder.
4. In the Open-Realty® Admin section, go to the Site Configuration. On the **Editor / HTML** tab you must edit the setting for the **WYSIWYG Editor** option and select the editor that you just uploaded.

**TIP:** There should not be a second folder of the same name as your editor like:

`/include/class/fckeditor/fckeditor`

Some extraction software will extract the compressed files and folders to a folder of the same name as the archive thus putting your actual WYSIWYG editor folder inside a folder of the same name.

# UPGRADING OPEN-REALTY®

This section is for users who already have an existing website using Open-Realty® 1.1.5 or 2.x and wish to upgrade to the latest version. The Open-Realty® development team strongly urges all users to keep their Open-Realty® installation upgraded to the latest version of the software for the latest bug fixes, security enhancements and new features.

---

## Getting Started

***BACKUP!*** Always backup your existing website and database prior to attempting an upgrade. Backing up your database can be done using your favorite database management tool which is often available through your hosting provider's control panel (Usually phpMyAdmin).

**Important!** If you are upgrading from a version prior to version 2.1, make sure your existing site's setup includes the 'type' listing field and that all of your listings are assigned a type. Failure to do this will result in the upgrade script not being able to properly convert your site to the new Property Class System.

1. [Download](#) the latest official Open-Realty® release. You can also use SVN to get the latest work in progress. SVN Revisions are not official releases and may have bugs in them.
  1. Go to <http://www.open-realty.org/download.html> to download Open-Realty®.
2. Extract the downloaded file(s) and upload to your website, either to the root folder on your site, usually called www or public html, or to a sub-folder of your root.
  1. **NOTE REGARDING TEMPLATES:** If you modified one of the default Open-Realty® templates but did not change the folder name of the template to something unique, uploading the new version of Open-Realty® will overwrite your custom template changes. Either rename your custom template prior to uploading or do not upload the default template of the same name.
3. Set Permissions (chmod) on the following files/folders so that they are writable by the web server. You can set permissions(chmod) using most FTP Clients, Web based File managers, or via a shell account. All of the following files/folders should have permissions of 777 or 755 depending on your host's configuration.
  1. `include/common.php`
  2. `images/listing_photos`
  3. `images/user_photos`
  4. `images/vtour_photos`
  5. `images/page_upload`
4. Create a blank MySQL or postgres database on your server. Most hosting providers provide a control panel interface for doing this, you should check with your hosting provider if you are not sure how to do this. You should also create a database user and assign them All Privileges to the database you created. Write down the database name, username, and password you just created.
5. Using your web browser, go to `http://www.yourdomain.com/installpath/install/index.php`. For example if you installed this on a domain called `myrealestatecompany.com` in a sub-directory called "open-realty" you would go to, `http://www.myrealestatecompany.com/open-realty/install/index.php`.

## Running the installer

1. Read the license agreement and click the "Agree" button to continue.
2. Select the default language for your site and click submit. **NOTE:** English is the only language file that is regularly maintained by the Development team. If you have difficulty installing due to blank pages in another language it is likely due to an incomplete or missing lang file for your selected lang. To fix that you can open the /install/lang/en/ folder and replace the lang.inc.php file from your configured lang folder with the one from the 'en' folder.
3. The next screen verifies the permissions you set in step 4. If it finds any permission problems correct them and refresh the page. When all permissions are correct; Click "Click to Continue Installation".
4. You now need to select the type of installation you are doing. You should be selecting one of the two Upgrade options. The options are as follows.
  1. New Install Of Open-Realty 2.x
  2. Upgrade from Open-Realty 2.x (Beta 1 or higher) **BACKUP your site and database before you continue.**
  3. Upgrade from Open-Realty 1.1.5 **BACKUP your site and database before you continue.**
  4. Update Path and URL information only - This option should be used when you have moved your open-reealty install to a new directory.
5. Select the type of database you created in step 4 of "Getting Started" above. Insert your database information from step 5 into the appropriate fields. The last four settings should not be changed unless you know what you are doing. Click next.
6. This step creates the common.php file, which stores the database configuration information for Open-Realty® to use. Click "Continue to setup the database."
7. This step will create and populate the Open-Realty® database tables.
8. Click "Click here to configure your installation". Your login information will be unchanged from the prior version you had installed.

## Search Engine Friendly URLs and Optimization

Open-Realty® has the ability to be configured to use Search Engine Friendly (SEF) URLs. These are URLs that appear in the user's browser, and search engines, as a standard html file name instead of the often cryptic URLs in PHP scripts that include variables, actions and values.

To enable SEF URLs there are two steps that are required as follows:

1. Change the name of the included ".htaccess-or" to ".htaccess" on your server. This file contains the rules that are used by the Apache rewrite engine.
2. In the Site Configuration on the **SEO** Tab edit the setting for the **URL Type** option and set it to "**Search Engine Friendly**".

Please read the **Administrators Guide** for more information on how to further configure and customize Open-Realty®.

Congratulations your Open-Realty® installation is complete! You must delete the install directory before your site will be accessible.

**TEMPLATE UPGRADE NOTE:** Since the release of your previous version there may have been some changes to the templates, including the addition of new template files. Compare

your custom template with one of the included templates for new template files and copy any missing template files to your custom template.



# MOVING TO A NEW HOST

This section is for users who already have an existing website using Open-Realty® and are moving their website to a new host.

---

## Getting Started

1. ***BACKUP!*** Always backup your existing website and database prior to attempting to move to a new host. Backing up your database can be done using your favorite database management tool which is often available through your hosting provider's control panel (Usually phpMyAdmin). Backing up your files can be done by downloading ALL of the Open-Realty® files from your current host to your local computer using your favorite FTP client (For example Filezilla).
  1. **TIP:** Some hosts provide a site backup option through the control panel that will let you download a single, archive file that includes your database backup file and all of the files from your site. Also, check with your new host as they may provide an option or method to automatically transfer your existing site and database to the new host. If you have this option, use it and skip to step X.
2. Using your favorite FTP client (For example Filezilla), upload ALL of your Open-Realty® files to the new host.
3. Create a new database, database user and give the user full permissions on the database. WRITE DOWN this information, you will need it later.
4. Use phpMyAdmin, or another available database management tool to restore the database backup to the new, blank database you just created on the new host.
5. EDIT: '*include/common.php*' which should already be uploaded to your new host. Replace the old database information with your new database information that you created in step 3. Modify the following lines as needed:
  1. `$db_user = "mydbuser";`
  2. `$db_password = "mydbpassword";`
  3. `$db_database = "mydatabase";`
  4. `$db_server = "localhost";`
6. UPLOAD: Upload the /install folder from your Open-Realty® version. If you no longer have the /install folder for your version, or are using an old version you can download the current version of Open-Realty® and then follow the upgrading instructions in the "UPGRADING OPEN-REALTY®" section to upgrade your Open-Realty® installation while moving it.
7. Using your web browser, go to <http://www.yourdomain.com/installpath/install/index.php>. For example if you installed this on a domain called myrealestatecompany.com in a sub-directory called "open-realty" you would go to, <http://www.myrealestatecompany.com/open-realty/install/index.php>.

## Running the installer

1. Read the license agreement and click the "Agree" button to continue.
2. Select the default language for your site and click submit, this **MUST** be the same language you were using on your old host. Click Submit.
3. The next screen verifies the permissions you set in step 4. If it finds any permission problems correct them and refresh the page. When all permissions are correct; Click "Click to Continue Installation".
4. You now need to select the type of installation you are doing. You should be selecting "Update Path and URL information only - This option should be used when you have moved your Open-Realty install to a new directory. The full options are as follows.
  1. New Install Of Open-Realty 2.x
  2. Upgrade from Open-Realty 2.x (Beta 1 or higher) **BACKUP your site and database before you continue.**
  3. Upgrade from Open-Realty 1.1.5 **BACKUP your site and database before you continue.**
  4. Update Path and URL information only - This option should be used when you have moved your open-realty install to a new directory.
5. Verify that the installer is using the correct settings for your database and paths. Click next.
6. This step creates the common.php file, which stores the database configuration information for Open-Realty® to use.

Please read the **Administrators Guide** for more information on how to further configure and customize Open-Realty®.

Congratulations your Open-Realty® installation is complete! You must delete the install directory before your site will be accessible.

# OPEN-REALTY® ADMINISTRATION

Welcome to the Open-Realty® Admin area! If you just installed Open-Realty® it is recommended that you edit your site's configuration to customize the settings to your needs and make the site yours. You will find information on all of the Site Configuration settings below in this section of the documentation.

If you have edited your configuration to your needs and tastes, you now have your Open-Realty® site setup and ready to use with a default configuration and set of data! You can add agent accounts, add listings or continue to customize Open-Realty® further to suit your needs. One of the first things many people want to do is install a custom template or create their own custom template. Templates are covered in the Template Documentation section of this documentation.

Static content pages on your Open-Realty® website, such as the home page, can be edited through the Page Editor in the Admin area.

Now you can start to define how you want your listings to be setup. The first step is to make a list of the fields you want each listing to be able to have. You need to remove any of the existing fields you don't want, edit the content of existing fields to suit your needs, and create any new fields using the Listing Template Editor.

Once your listing fields are setup, you should setup any fields you want to have for agents. This work similar to the method used above to edit your listings template. See the Agent Template Editor section for more information.

The last set of fields you need to setup are the Member fields. These are what users will have to fill in to sign up to save searches and favorite listings. Again, this the same process now using the Member Template Editor.

Finally, go take a break, you deserve it!!!!

Next, you need to create agent accounts for any agents you need using the User Manager. If you turned on agent sign up in the Site Configuration agents can create their own accounts, and you don't need to do this for them (you must place a link to the Agent Sign up on your Open-Realty® site, see the Template Tag documentation for a template tag that will do this).

Now that the agents are created, you can either start Adding Listings for them, or let them log in and add their own listings. Also, I suggest reading some more information on the Virtual Tours.

# Configuration

Open-Realty® is a very diverse and powerful application and is highly customizable through the Configuration area. Here you can make numerous changes that will effect the look, feel and function of your Open-Realty® website.

Due to all the options, the configuration section is a bit large, but we have broken it down into tabbed sections for easier navigation. If you take the time to read this document and follow all the steps you will find that it is a fairly quick and mostly painless task.

Each tab you see General, Template, SEO, etc; needs to be configured to match your needs for your site. Please review the details of each section and make any appropriate changes to your setup.

## General

### General Info

**Admin Name** - This should be the name you want to appear on any emails sent from the site.

**Admin Email** - This is the email address that all messages will be sent from.

**Site Email** - Optional email address to use as the sender for site emails. If left blank, the admin email address will be used.

**Company Name** - This is your company name, it can be used on any page by referencing the {company\_name} template tag.

**Company Location** - This is your company location, it can be used on any page by referencing the {company\_location} template tag.

**Company Logo** - This is the location of your company/site logo on your web server, it can be used on any page by referencing the {company\_logo} template tag.

**Automatic Update Check** - Open-Realty® will automatically check for available updates to your installed version. By default this is turned off to keep the control panel as fast as possible. Turning it on will make Open-Realty® check with Open-Realty.org for available updates and may slow the control panel down slightly.

**Demo Mode** - This will disable users from editing their passwords or any of the default Open-Realty® pages and the Site Configuration.

**Maintenance Mode** - This will enable maintenance mode which will display the maintenance mode page to all users except the "**admin**" user. The maintenance mode page layout and content is controlled by the "maintenance\_mode.html" template file in the site template folder. The site admin will see the site as normal so they can test any changes they are making before making them live for site visitors.

### Server Paths

**You CAN'T change these settings.** These settings are hard coded and can NOT be changed from the site configuration.

If your site is moved to a new directory or server you will need to run the install program again and select "**Update Path and URL information ONLY**"

**Base URL** - This is the website URL where your Open-Realty® install is located. Leave off the trailing slash.

**Base path** - This is the actual location of the Open-Realty® install on your web server. Leave off the trailing slash.

## **Language Settings**

**Default Language** - This is the configured language. *This setting is disabled until multi-language functions are completed*

## **Template**

### **Template Settings**

**Character Set** - This is character set that will be used for the text encoding on the site. **WARNING** Changing the Character Set after you have your site running with live data can cause unexpected results with existing data that may have been entered. Special characters entered under one character set may not display properly under a different one.

**Site Template** - This is the Template that the site will use. The Default included template is named "**lazuli**". Any new template directories in the /template directory will appear in this section.

**Administrative Template** - This is the Template for the Admin section of the site. New templates for this will be added to /admin/template and will appear in this section.

**Listing Template Sections** - This is where to define all template sections for the listing template that you want users to be able to map fields to. *Be careful about removing items from this list if you have fields mapped to that template section already*

**Listing Template** - This is the template used for the layout of the listing display. These templates are placed in your template directory and are named "listing\_detail\_templatename.html"

**Search Result Template** - This is the template used for the layout of the search results. These templates are placed in your template directory and are named "search\_result\_templatename.html"

**View Agent Template** - This is the template for the View Agent page.

**VTour Template** - This is the template that will be used for the VTour pop-up windows.

**Listing Notification Template** - This is the template used to send members notification of new listings that match their saved searches. The notifications are processed through a cron job. See the "Automating Tasks" section of this documentation for more details on setting up the Listing Notification system.

## SEO (Search Engine Optimization)

### Search Engine Optimization Settings

**URL Type** - This is used to select standard or search engine friendly URLs. This is used in conjunction with the .htaccess-or file that's included in the base directory.

**If you want to use Search Engine friendly URLs you need to change the name of .htaccess-or to .htaccess and upload it to the server. Your server must also allow URL Rewriting.**

**Space Character** - If SEO URLs are turned on this is the character that will replace spaces. Default will replace with a +, a hyphen is argued to be more search engine friendly. If Hyphen is used, you MUST use the .htaccess rewrite rules from version 2.5

**Default Page Title** - This is the default page title for your site. This will be the title on your home page and all pages other than the listing detail pages unless it is a page editor created page and you specify a custom title for that page in the page editor.

**Default META Keywords** - These are the default keywords that Open-Realty® will place into the META keywords tag for your site. These will be used for all pages other than the listing detail pages unless it is a page editor created page and you specify a custom META Keywords for that page in the page editor.

**Default META Description** - This is the default description that Open-Realty® will place into the META description tag on your site. This will be used for all pages other than the listing detail pages unless it is a page editor created page and you specify a custom META Description for that page in the page editor.

**Listing Page Title** - This is the title for your listing detail pages. You can use normal text here or place listing data into the title by using the SEO Listing Template Tags (See the Listing Template Tags section of the documentation). You can use a combination of text and Template tags in this field.

**Listing META Keywords** - These are the keywords for the META keywords tag for your listing detail pages. You can use normal text here or place listing data into the keywords by using the SEO Listing Template Tags (See the Listing Template Tags section of the documentation). You can use a combination of text and Template tags in this field.

**Listing META Description** - This is the description for the META description tag for your listing detail pages. You can use normal text here or place listing data into the description by using the SEO Listing Template Tags (See the Listing Template Tags section of the documentation). You can use a combination of text and Template tags in this field.

## Editor / HTML

### WYSIWYG Editor Settings

**WYSIWYG Editor** - This is to select which WYSIWYG editor to use in the page editor section of Open-Realty®.

**Show Edit Link on frontend?** - Should we show the Edit Page link to Admins on front-end pages of the site?

**Execute PHP Code?** - If this is turned on any PHP code entered into the WYSIWYG editor will be run. Doing this is a BAD IDEA unless you trust everyone that has page editor abilities. USE AT YOUR OWN RISK!. When entering the PHP command, you need to enter it in code view on the editor and surround it in html comments.

For Example, if you are trying to enter:

```
<?php echo 'text'; ?>
```

You need to enter it as:

```
<!--<?php echo 'text'; ?>-->
```

Note that there are no spaces between the html comment marks and the PHP opening/closing tags.

**MBString is enabled at the server?** - By default it is set to "No". Check with your hosting support to see if MBString (MultiByte) is enabled at the server. If this is enabled set to "Yes". MBString support required if you are going to store "special characters" (Multilingual Feature) at the Database (using the Listing/Page/Blog Editors with or without a WYSIWYG Editor).

## **HTML Settings**

**Add Linefeeds** - Should Open-Realty® convert returns into line feeds?

**Strip HTML** - Should any html code in listings be removed?

**Allowed HTML** - If you are stripping the HTML which html tags can a person input

## **Numbers**

### **Number Formatting**

**International Number Format** - Support for international numbering format

**Decimals for numbers** - Number of decimals to show for number fields

**Decimals for prices** - Number of decimals to show for price fields

**Force Display of Decimals** - By default, if a number is a whole number we will not show decimal places. If set to "yes", then we will force the display of decimal places even on whole numbers.

**Money Format** - Defaults to \$123, but others use different formats

**Money Sign** - Default is dollars (\$), but it could be £(&#163;) for pounds or €(&#128;) for euros

**Date Format** - Controls the display format of dates on the site.

**Blank Price Text** - Should Open-Realty® replace blank price values with "Call For Price" language variable? This variable can be changed in your lang.inc.php file if you wish to use different text.

**Price Field** - Select the listing field that contains the listing price.

## Uploads/Images

### Upload Settings

**Allowed Extensions** - Allowed file extensions for uploads. Add additional extensions here as required, some extensions may not be an allowed file type so be sure to check the file type and update that as needed.

**Allowed file types** - Allowed file types for uploads. These are MIME file types.

**Create Thumbnails** - Would you like to create thumbnails of uploaded images?

**Thumbnail width** - The width in pixels that thumbnails will be created at. (Keep in mind formatting of the default template was created with the default width, some formatting could be thrown off by changing this)

**Thumbnail height** - The height in pixels that thumbnails will be created at. (Keep in mind formatting of the default template was created with the default width, some formatting could be thrown off by changing this)

**Display Thumbnails by** - Which dimension, (width, height, both) should we display thumbnails with? If create thumbnails is turned on this is the dimension that we will use to create thumbnails with.

**Image Tool** - Which image editing tool would you like to use? Default is GD libs as it is installed and compiled into most Apache/php web servers. However you can also use ImageMagick in case you prefer it or don't have GD Libs.

**ImageMagick Path** - This is the path to your ImageMagick installation on your server. This is only used if you have set the image tool to ImageMagick.

**JPEG Quality** - Used with GD libs only. This is the quality of JPEG images created with GD Libs. Smaller number gives smaller file size and lower quality, higher number is larger filesize and higher quality.

**Resize Image** - Would you like uploaded images to be resized? If set to "Yes" then uploaded images will be resized to their Maximum Image Width for their image type as defined in Upload Limits

**Resize By** - What to use when resizing images:

- Width - Use the image width setting to resize the image
- Height - Use the image height setting to resize the image
- Best Fit - Uses both the width and height settings to determine which setting to resize by in order to make the resized image fit within both the maximum width and maximum height settings
- Both - Resizes by both the maximum width and height settings. THIS WILL LIKELY CAUSE DISTORTION UNLESS YOUR IMAGE AND THE WIDTH AND HEIGHT SETTINGS ARE THE EXACT SAME ASPECT RATIO.

**GD Version 2.x** - Are GD Libs version 2.x installed on the server? Default is YES as most everybody should be using 2.x by now, if you happen to have 1.x versions choose "No".

**Use No Photo** - If a listing doesn't have photos uploaded for it, Should Open-Realty® display the /images/nophoto.gif instead?

### Upload Limits



**Max # of images per listing** - Maximum allowed number of images that can be uploaded per listing.

**Max File Size** - Maximum allowed file size in bytes for uploaded listing images.

**Max Image Width** - Maximum image width for uploaded listing images. If "Resize Image" is set to "Yes" in the **Upload Settings** and uploaded images are larger than their "Maximum Image Width" setting, then uploaded listing images will be automatically resized to their Maximum Image width.

**Max Image Height** - Maximum image height for uploaded listing images. If "Resize Image" is set to "Yes" in the **Upload Settings** and uploaded images are larger than their "Maximum Image Width" setting, then uploaded listing images will be automatically resized to their Maximum Image height.

**Max # of images per user** - Maximum allowed number of images that can be uploaded per user.

**Max File Size** - Maximum allowed file size in bytes for uploaded user images.

**Max File Width** - Maximum image width for uploaded User images. If "Resize Image" is set to "Yes" in the **Upload Settings** and uploaded images are larger than their "Maximum File Width" setting, then uploaded listing images will be automatically resized to their Maximum Image width.

**Max File Height** - Maximum image height for uploaded user images. If "Resize Image" is set to "Yes" in the **Upload Settings** and uploaded images are larger than their "Maximum File Width" setting, then uploaded listing images will be automatically resized to their Maximum File height.

**Max # of Virtual Tours per listing** - Maximum allowed number of virtual tours that can be uploaded per listing. (Note: only 1 EGG Solution type of virtual tour can be uploaded per listing at any time)

**Max File Size** - Maximum allowed file size in bytes for uploaded virtual tours.

**Max Image Width** - Maximum image width for uploaded Virtual Tours. The "Resize Image" setting will NOT affect virtual tours.

## **Image Display**

**Main Image Display by** - How to display the Main Image called with the main image or main image java tags?

**Main Image width** - Main Image display width.

**Main Image height** - Main Image display height.

**Number of Columns** - The number of columns to use on the Java Listing Images by Rows template Tag before starting a new row.

## **Uploads/Files**

### **File Upload Settings**

**Allowed Extensions** - Allowed file extensions

## **File Upload Limits**

**Max # of files per listing** - Max # of files for a given listing

**Max File Size** - Max File Size in bytes for listing files.

**Max # of files per user** - Max # of files for a given user

**Max File Size** - Max File Size in bytes for user files.

## **File Display Settings**

**Show the File Icon** - Sets if the file icon should be displayed. If this is set to yes, Open-Realty® will display an icon image from the /files/ folder where the image's name is the file extension of the file. Open-Realty® expects that there will be a matching file for all allowed extensions.

**File Display Option** - Should we display the file caption or the file name for the link? If caption is selected and it is empty defaults to filename

**Show File Size** - Should we display the file size?

**File Icon Width** - Default icons included with Open-Realty® are 16x16. However, if you use a different icon set you may want to change these sizes to the size of your new icons. All icons are expected to be the same size.

**File Icon Height** - Default icons included with Open-Realty® are 16x16. However, if you use a different icon set you may want to change these sizes to the size of your new icons. All icons are expected to be the same size.

## **Search**

### **Search Options**

**Max Min/Max Steps** - The max allowed steps in min/max searches. If your step value will create more steps than this maximum, a new step value will be calculated to stay within the max allowed steps to prevent memory overflow problems. Setting this to 0 will disable this protection

As an example, Lets assume you have a price field set to searchable with a min/max search and have a step value of 50000 (50,000) entered. If you have a listing in your database with no price, and one with a price of 1000000 (1,000,000) then the min/max search function will calculate and generate a range of 20 values for both the min and max search boxes for a total of 40 options. This doesn't take much time and won't be noticed. However, if a agent makes a typo while entering a listing, or you configure the step value too low things could get out of hand quickly. For example, say an agent entered 100000000 (100,000,000) in that same scenario. Open-Realty® would now need to calculate and render 2,000 search options for both the min and max search boxes for a total of 4,000 options. This will likely cause a memory overflow on the server, and if not, then would cause the end user to have to wait for 4,000 search options to be downloaded and rendered by their browser.

Having the Max Steps option set will first calculate how many step values would be calculated in a given scenario and if it is more than the max allowed, will override the settings with a more reasonable amount.

**Listings Per Page** - Number of listings to show on each Search Result Page.

**Sort Searches By** - The field you want your search results sorted by default. Only fields that are set to be displayed on browse are available for selection plus the option to sort randomly. *This should be a required field that exist in every class. Sorting by a field that does not have a value or does not exist on a listing or property class can result in Open-Realty® not displaying the listing, as the SQL server will not return it.*

**Sort Type** - How you want your searches sorted: ASC will sort lowest to highest, DESC will sort highest to lowest.

**Static Sortby Option** - Special Static Sortby option. If this is configured search results will **ALWAYS** first be sorted by this sortby before the default or user selected sortbys. This is useful primarily if you want to sort Featured Listings first in your search results

**Static Sortby's Sorttype** - If you have a Static Sortby Option selected this is the Sorttype for that Static Sortby option.

**Show Count in Search Options** - Would you like the number of listings shown in the search options?

**Maximum Search Results** - Maximum number of search results any search shall return. 0 = NO LIMIT

**Search form checkbox list separator** - Enter the html or text you want to use to separate each of the checkbox options.

**Short textarea characters** - How many characters to display on the text area short template tag?

## **VTours**

### **VTour Options**

**VTour Width** - The width in pixels of the VTour display.

**VTour Width** - The width in pixels of the VTour display

**VTour FOV** - The Field Of View of your VTour display window. This setting basically controls the initial Zoom of the VTour. This setting is in degrees with a Min of 12 and Max of 165.

**Popup Width** - Width in pixels of the VTour popup window

**Popup Height** - Height in pixels of the VTour popup window

## **Notify**

### **Notification Settings**

**Email User Notification** - Should the admin be sent an email notification when users register?

**Email Listing Notification** - Should the admin be sent an email notification when listings are added?

**Email Users Listing Notification** - Should the users be sent an email notification when listings are added?

**Email Users Registration Information** - Should the users receive an email notification with their registration info when they register?

**Use image verification on contact forms?** - Adds an image code to the contact forms. Users are required to enter the code in order to submit the form. Requires server to have the GD image libraries installed. *The content of these emails can be edited in the language file.*

**Disable HTTP Referrer Check on contact forms** - Disables HTTP Referrer Check on contact forms. NOT RECOMMENDED! This relies on using the Image Verification as a minimum security feature, if image verification is disabled, then the HTTP Referrer check will remain active.

**Include Senders IP in Email to Agent** - Should we include the senders IP address in the contact agent emails?

## Users

### Signup Settings

**Use image verification on signup page** - Adds an image code to the signup forms. Users are required to enter the code in order to submit the form. Requires server to have the GD image libraries installed.

**Require email address confirmation** - Setting this to yes will require that the user confirm their email address by clicking a link in the signup email.

### Member Permissions

**Moderate Members** - Do new members require the admin to approve them before they are active?

**Member Signup** - Are members allowed to signup?

### Agent Permissions

**Moderate Agents** - Do new agents require the admin to approve them before they are active?

**Agent Signup** - Are agents allowed to signup? **If you set this to 'Yes' you will want to place a link to your agent signup page somewhere on your site. The link is: *index.php?action=signup&type=agent* or use the documented template tag on your site to place the signup link: *{url\_agent\_signup}***

**Active** - Are new agents active by default?

**Admin** - Are new agents admins by default?

**Limit # of Listings** - This is the number of listings agents can create by default. (-1 is unlimited).

**Limit # of Featured Listings** - This is the number of listings that an agent can set to Featured. (-1 is unlimited).

**Display Order** - A ranking number for the ordering of agents on the agent list and user manager.

**Can edit other users** - By default can new agents modify other agents/member profiles?

**Can edit other agents listings** - By default, can new agents modify other agents listings?

**Feature listings** - Can new agents feature listings by default?

**Moderate listings** - Can new agents moderate listings by default?

**View site log** - Can new agents view log by default?

**Can edit site config** - By default, can new agents edit site config?

**Can edit member template** - By default, can new agents edit the member template?

**Can edit agent template** - By default, can new agents edit the agent template?

**Can edit listing template** - By default, can new agents edit the listing template?

**Export listings** - Can new agents mark listings as exportable by default? You must enable Allow Listing Exports in the Listing Section of the Site Config to use this option.

**Change listing expirations** - Can new agents change listings expirations by default?

**Edit pages** - Can new agents edit pages by default?

**Virtual Tours** - By default, Can new agents add virtual tours to listings?

**Files** - By default, Can new agents add files to listings?

**Blogging Privileges** - Does the agent have the ability to create blog posts?

**Can manage add-ons** - Can the agent access the add-on manager and manage add-ons?

## **Agent Settings**

**Agents per page** - Number of agents to display per page on the view agents page.

## **Listings**

### **Listing Settings**

**Allow Multiple Property Class Assignments** - Can listings be assigned to more than one property class? Default is NO

**Number of Featured Listings** - How many Featured listings should be displayed?

**Use Expiration** - Should Open-Realty® use expiration?

**Expire After** - How long should listings be active?

**Moderate Listings** - Should new listings require moderator approval in order to be Active

**Allow Listing Exports** - If enabled agents will be given the option of setting if a listing should be included in an MLS export or not. You need a custom export script currently to use this.

**Show Listedby Link For Admin's Listings** - Should we show the listedby link for listings owned by the site admin.

**Show NextPrev Bar On Listing Page** - Show the next\_prev bar on listing pages, to allow users to move between listing without having to go back to the search result page. Warning this will cause a small performance hit.

**Show Notes on listing editor** - Yes will show the notes field on the listing editor. No will make the notes field hidden but still exist.

**Feature list separator** - Enter the html or text you want to use to separate the items in the feature list.

## Maps

### Map Settings

**Use Map** - What Map system would you like to use? US Google is the default, Mapquest and Yahoo Maps are other choices.

**Address Field** - Name of the address field? This field will be the first address field.

**Address Field 2** - Second Address Field. This field will be appended to the end of the first address field. (This is used when storing street number/names/suffix/direction in separate fields)

**Address Field 3** - Third Address Field. This field will be appended to the end of the second address field. (This is used when storing street number/names/suffix/direction in separate fields)

**Address Field 4** - Fourth Address Field. This field will be appended to the end of the third address field. (This is used when storing street number/names/suffix/direction in separate fields)

**City Field** - Name of the city field?

**State Field** - Name of the State field?

**Zip Field** - Name of the Zip Code field?

**Country Field** - Your country field. (Only used if not using a country specific mapping choice.)

## Vcards

### **Agent Vcard Settings**

The agents first name, last name, and email address are pulled from their user account information. This information is sufficient to create the vcard. The below field can be used to include additional information in the vcard.

**Phone** - Name of the phone field?

**Fax** - Name of the fax field?

**Mobile** - Name of the mobile phone field?

**Address** - Name of the address field?

**City** - Name of the city field?

**State** - Name of the state field?

**Zip** - Name of the zip code field?

**Country** - Name of the country field?

**Notes** - Name of the notes field?

**Homepage** - Name of the homepage field?

## RSS

In order to setup links to your RSS Newsfeeds you will need to add a link on your site to one of the following actions:

**rss\_featured\_listings** - This will provide the user with an RSS Newsfeed of the featured listings. A link to this Newsfeed would look something like this: `<a href="index.php?action=rss_featured_listings" >{lang_rss_featured_link_title}</a>`

**rss\_lastmodified\_listings** - This will provide the user with an RSS Newsfeed of the latest modified listings. A link to this Newsfeed would look something like this: `<a href="index.php?action=rss_lastmodified_listings" >{lang_rss_lastmodified_link_title}</a>`

**NOTE:** RSS Feeds may break in rare circumstances when there are special characters in the Listing Title field and Multi-Byte String support is not enabled in PHP. There should not be any problems without Multi-Byte String support as long as the database collation and Open-Realty® are both set to UTF-8 character sets.

### **RSS Configuration**

Here is the configuration tab for configuring your site's RSS Newsfeeds.

**Featured Listing Feed Title** - This is the title to use for the RSS feed of featured listings.

**Featured Listing Feed Description** - This is the description to use for the RSS feed of featured listings.

**Featured Listing Description** - This is the description to use for the individual listings in the featured listings feed. You can include HTML and use any of the listing template tags.

**Number of Featured Listings** - This is the maximum number of featured listings that will be placed in the feed. 0 = no limit

**Last Modified Feed Title** - This is the title to use for the RSS feed of last modified listings.

**Last Modified Listings Feed Description** - This is the description to use for the RSS feed of last modified listings.

**Last Modified Listings Description** - This is the description to use for the individual listings in the last modified listings feed. You can include HTML and use any of the listing template tags.

**Number of Modified Listing** - This is the maximum number of modified listings that will be placed in the feed.

## Help Links

### Help Link Configuration

**Use Help Link** - This option is to display a help link at the top of the admin template. Currently the default links are to the old Wiki Documentation project.

## Blog

### Blog Configuration

**Moderate comments** - This option determines if new blog comments require moderator approval before they are visible.



# Page Editor

The Page editor, is a HTML editor for editing and adding pages to your site that are not directly tied to listing information. If you installed one of the available WYSIWYG (*What You See Is What You Get*) HTML editors during the installation then editing the content of pages will be more simplified as you will see your page content how it will appear on the site.

The content of the index (Home) page is controlled by the Open-Realty® Page Editor. In addition the default installation also has three additional pages that are Page Editor created pages, those pages are the About Us, Contact Us and the Legal Page.

There are three different WYSIWYG editors available, see the Installing Open-Realty® section for download and installation information. There are also several configuration options available for the Page Editor, see WYSIWYG Configuration for more information.

## The Page Editor Screen:

**Edit Page** - Select the Page Title you wish to edit and select **Edit Page**

**Using PHP in a page** - To insert PHP into the WYSIWYG pages, you first must turn on Execute PHP in the WYSIWYG Configuration. Then when entering the PHP command, you need to enter it in code view on the editor and surround it in html comments. For Example, if you are trying to enter "`<?php echo 'text'; ?>`" you need to enter "`<!--<?php echo 'text'; ?>-->`". Note that there are no spaces between the html comment marks and the PHP opening/closing tags.

**Create New Page** - Click on this to Create a new page.

**Title** - The Title of the page - Will be the full title of the page, will not append to the configured default page title.

**Meta Description** - The Meta Description for this page.

**Meta Keywords** - The Meta Keywords for this page.

**Template tag for page** - This is the template tag for the page you are editing. Use the template tag in your templates to link to the page you are working on.

**Link to this Page** - This is a link to the page, in case you wish to link from an external source to the page you are working on.

**Page Editor Window** - In this area the page editor you selected in the WYSIWYG Configuration will be displayed. Usage of each editor would be too extensive to include here but most of it is pretty standard WYSIWYG editor stuff.

**Submit** - Save your changes

**Delete Page** - Delete the page you have selected from the database.

# Blog Editor

The Blog editor, is a HTML editor for editing and adding blog posts to your site. If you installed one of the available WYSIWYG (*What You See Is What You Get*) HTML editors during the installation, then editing your blog posts will be more simplified as you will see your blog post exactly how it will appear on the site.

There are three different WYSIWYG editors available, see the Installing Open-Realty® section for download and installation information. There are also several configuration options available for the Blog Editor, see WYSIWYG Configuration for more information.

## Blog Manager (Navigation View)

The main screen of the Blog Manager is the navigation view which displays all of your blog posts and details about them. At the top of the page is several filtering options for displaying All blogs, Published blogs, Drafts and blogs that are require review. The window will display a table listing all of your blog posts with the following details:

- **Post** - Shows the title of the blog post. Clicking on the post will take you to the Blog Editor.
  - **Author** - Shows which user created the blog post.
  - **Keywords** - Displays keywords associated with the blog post
  - **Comments** - Shows the number of comments on the blog post. Clicking on this number will take you to the Edit Comments page of the Blog Manager
  - **Date** - Displays the data the blog post was created
  - **Publication Status** - Displays the status of the blog post. Statuses are:
    - **Published** - The blog post is published and active
    - **Draft** - The blog post is not active and still in draft mode
    - **Review** - The blog post requires review before it will be published
- 

## Blog Manager (Editor View)

**Title** - The title for your blog post.

**Template tag for blog** - This is the template tag for the blog post you are editing. Use the template tag in your templates to link to the blog post you are working on.

**Link to this blog** - This is a link to the blog post, in case you wish to link from an external source to the blog post you are working on.

**Blog Editor Window** - This is the main post editor window. This is where all the content of your blog post will go. If you selected a WYSIWYG editor in the site configuration it will be used for editing your post. Usage of each editor would be too extensive to include here but most of it is pretty standard WYSIWYG editor stuff.

**Publication Status** - Displays the status of the blog post. Statuses are:

- **Published** - The blog post is published and active
- **Draft** - The blog post is not active and still in draft mode
- **Review** - The blog post requires review before it will be published

**Search Engine Optimization** - These fields are for search engine optimization and will be displayed on the blog post to help with search engine ranking.

- **Meta Description** - The Meta Description that will be used when viewing the blog post
- **Meta Keywords** - The Meta Keywords that will be used when viewing the blog post

**Submit** - Save your changes

**Delete Blog Entry** - Delete the blog that you are editing. **WARNING:** This can not be undone

---

## Comments

Clicking on the number of comments in the Blog Manager navigation will display all of the comments that have been posted for that blog post.

# Listing Template Editor

The Listing Template Editor is where you can edit and setup the fields that are used on your listings, the field content, field types, field orders and search setup for the fields. There are several options on this page:

- [Edit Field](#) - Using the drop down menu, select an existing listing field and click edit to edit the listing field
- [Add Field](#) - Click on this option to add a new listing field
- [Set Field Order](#) - Click on this option to Set Field Order
- [Search Setup](#) - Click on this option to Setup the Search
- [Search Results Setup](#) - Click on this option to Setup the Search Results

## Add or Edit Listing Fields

This section is for adding or editing Listing Fields. There are numerous options available for your listing fields to customize how they display on listings, search pages, search results etc. See the information below for setting up your listing fields.

### General Options

**Field Name** - This is the name of the field as it will be stored in the database.

**Field Type** - The Type of field that this is. Field Type effect the Add listing and Edit listing forms that agents fill out. (Text, Textarea, select list, select box, etc....)

- **Text** - Provides agents with a standard text box to enter data into.
- **Textarea** - Provides agents with a larger text box to enter data into. Use full for things like remarks.
- **Select** - Provides agents with a drop down list based on the field elements, from which they can select one option.
- **Select Multiple** - Provides agents with a drop down list based on the field elements, from which they can select multiple options.Option Box
- **Option Box** - Provides agents with a list based on the field elements with radio options next to each item, from which they can select one option.
- **Check Box** - Provides agents with a list based on the field elements with radio options next to each item, from which they can select multiple options.
- **Divider** - Creates a divider "-----" in place of the field to help break up a edit listing or add listing page's layout. *No real data can be entered into this field.*
- **Price** - Provides agents with a standard text box to enter a price into. Price should be entered without any money sign, commas, or decimals. *The money sign will prefix the text box in the add listing page and ".00" will suffix it. This also tells the system to format the price as setup in the site config for all listing display pages.*
- **URL** - Provides agents with a standard text box to enter a URL into. Agents should enter the full url including http:// the system will then create a hyperlink for then on the listing pages with from this URL.
- **Email** - Provides agents with a standard text box to enter an email address into. Email address will be clickable links on the listing page. NOT RECOMMENDED TO USE, as spam bots can easily read thsi address from your page and start spamming the email address entered.
- **Number** - Provides agents with a standard text box to enter a number value into. Will be formatted with decimal points as setup in site config.

- **Decimal** - Provides agents with a standard text box to enter a decimal value into. Same as number, except meant for values that will always have a decimal value.
- **Date** - Provides agents with a standard text box to enter data into, will show and enforce the required date format as defined in the site config.
- **Latitude** - Provides agents with a standard text box to enter a latitude into. There can only be one latitude field defined. It is used by the search engine when doing distance searches and by some mapping add-ons such as [TransparentMaps](#).
- **Longitude** - Provides agents with a standard text box to enter a longitude into. There can only be one longitude field defined. It is used by the search engine when doing distance searches and by some mapping add-ons such as [TransparentMaps](#).

**Required** - Is this field required to be filled out?

**Field Caption** - The Caption for the field that will be displayed to users.

**Field Elements** - The choices for multiple selection Field Types such as select lists, option boxes etc. Each choice should be separated by double pipes (||) Do not use spaces or returns after each option, run them all together as one line. For example: Option 1||Option 2||Option 3||Option 4

**Default Text** - The Default text to be displayed in this field

**Field Tool Tip** - Add a tool tip to the field that will display a question mark icon next to the field in the listing editor/add listing page so users can click on the icon and obtain information/tips regarding completing the field.

**Maximum Field Length** - Configure the maximum number of characters that can be entered into the field.

**Field Tool Tip** - Adds a tool tip to the add/edit listing pages for the field. (note: requires new css settings. If upgrading or using a custom template, you can copy the a.tooltip... classes from the vertical-menu style.css file.)

**Maximum Field Length** - Maximum number of characters that can be entered into this field. Applies to text, price, url, email, number, latitude, longitude, and decimal field types.

**Show Field to** - Use this option to restrict who will see this field. If you have the field set to display on your site using one of the template areas or individual field rendering template tags then this will determine who will see this field. Available options are:

- All Visitors - Everybody will be able to view this field.
- Members and Agents - Only registered users of your site can see this field.
- Agents Only - Only Agents will be able to view this field.
- Admin Only - Only Admins will be able to view this field on the front end of the site.

**Available in the following property classes** - Use this option to select which property classes this field will be available in. You can select multiple classes by clicking and dragging or by holding down the "Ctrl" key while clicking.

## **Listing Page Options**

**Location on Listing Page** - On the listing view page, the section of the template this field will appear in.

**Field Order on Listing Page** - The rank of this field on the Listing view page, the lower the number the higher on the list it will appear

## **Search Page Options**

**Allow Searching** - Check this option to make the field searchable

**Field Order on Search Page** - The rank of this field on the Search page, the lower the number the higher on the list of options it will appear

**Search Label** - The Label for this field that will appear on the Search Page.

**Search Type** - The type of search field that should be used for this field.

- **Partial Text Match** - This search option presents the user with a text box to enter text into. It will then find any listings where this field contains the entered text.
- **Option list of individual values** - Builds a option list, that allows multiple selections, based on the field elements for this field. This option requires that a listings have ALL the selected options to be returned in the search results.
- **Option list of individual values (OR search logic)** - Builds a option list, that allows multiple selections, based on the field elements for this field. This option will return a listings has ANY of the selected options.
- **CheckBox list of individual values** - Builds a list of options from the field elements with checkboxes next to each item. This option requires that a listings have ALL the selected options to be returned in the search results.
- **CheckBox list of individual values (OR search logic)** - Builds a list of options from the field elements with checkboxes next to each item. This option will return a listings has ANY of the selected options.
- **Pull down list of individual values** - Builds a pull down list, that allows a single selection, based on the field elements for this field.
- **Distinct list - allow multiple** - Builds a option list, that allows multiple selections, based on the actual field values for the existing listing. This option requires that a listings have ALL the selected options to be returned in the search results.
- **Distinct list - allow multiple (OR search logic)** - Builds a option list, that allows multiple selections, based on the actual field values for the existing listing. This option will return a listings has ANY of the selected options.
- **Pull down menu, 1 selection only** - Builds a pull down list, that allows a single selection, based on the actual field values for the existing listing.
- **Checkbox list of distinct values** - Builds a list of options, based on the actual field values for the existing listings, with check boxes next to each item. This option requires that a listings have ALL the selected options to be returned in the search results.
- **Checkbox list of distinct values (OR search logic)** - Builds a list of options, based on the actual field values for the existing listings, with check boxes next to each item. This option will return a listings has ANY of the selected options.
- **Radio button list of distinct values** - Builds a list of options, based on the actual field values for the existing listings, with radio options next to each item. Radio Options allow only a single selection to be made.
- **Two pull downs for min/max ++** - Builds two drop down boxes, one for the user to select the minimum value from and one drop down for the maximum value. Builds the drop down by looking at the minimum value in the database and the maximum and then placing incremental values based on the Step value specified by the user. For Example if the minimum value is 1 and the maximum is 10 and you specify a step value of 2 you will get drop downs with the following options. 1,3,5,7,9,10
- **Date range; two text fields** - Provides two text boxes to allow user to enter two dates. Will return any listings where the date for this field is between the dates entered by the user.
- **Single/Exact Date Search** - Provides a text boxes to allow user to enter a date. Will return any listings where the date for this field matches the date entered by the user.
- **Search For Empty(NULL) Values** - Provides a check box, if check will return only listings where there is NO value entered for this field.
- **Search For Non Empty Values** - Provides a check box, if check will return only listings where there is a value entered for this field.

**++ Step by** - If you used a range selection for the **Search Type** then this is the value to step each available option in the range by.

**NOTE:** The Step values can be manually defined instead of using a **Step By** value that then builds the values based off the values stored in the listing database. In the Step By field, define your Step By using this format:

- MIN\_VALUE|MAX\_VALUE|STEP\_VALUE

For example, using "0|300000|20000" will start with 0 and go to 300,000 in steps of 20,000.

## **Search Result Options**

**Display on Search Results page** - Check this option to make the field show up on the search results page.

**Field Order on Search Results Page** - The rank of this field on the Search page. It corresponds to the {field\_#} tag in the search result template. So the field with a rank of 1 will be displayed in the {field\_1} spot. Having multiple fields with the same rank will cause problems.

## **Set Field Order**

This page is to edit the field order for all of your listing fields at one time. It will display your fields in the **Location on Listing Page** they are currently assigned to. You can edit the **Field Order on Listing Page** value to raise or lower a field's position on the listing view page. Click **Set Order** to save your changes.

## **Search Setup**

This page is to edit the Search Setup for all of your Listing Fields at one time.

### **Search Page Options**

**Allow Searching** - Check this option to make the field searchable (If you Allow Searching you **MUST** define a **Search Type**)

**Field Order on Search Page** - The rank of this field on the Search page, the lower the number the higher on the list of options it will appear

**Search Label** - The Label for this field that will appear on the Search Page.

**Search Type** - The type of search field that should be used for this field. (If you Allowed Searching you **MUST** define a **Search Type**)

*Multiple option searches search using 'AND' logic requiring all of the selected options in the results. There are several search type options that specify 'OR' logic which will only require that one of the selected options be in each result.*

**Step by** - If you used a range selection for the **Search Type** then this is the value to step each available option in the range by. (Do not set this value too small or the script will timeout or run out of memory trying to generate all the step options for the search)

## **Search Results**

This screen is a single screen to edit the Search Results Setup for all of your Listing Fields.

### **Search Results Options**

**Field Name** - Displays the name of the field. Also displays the display name of the field in quotes

**Display On Search Result Page** - If this field should be displayed on the search results page.

**Field Order on Search Result Page** - The order of the fields being displayed on the search results page.



# Property Classes

The Property Class Editor is where you can add or edit the property classes that are available. Property classes allow you to define fields in the Edit Listing Template and assign them to specific property classes. As an example if you were listing both cars and houses for sale you can have a class for Homes and one for Cars and assign fields such as Make, Model to the Car class and Sq Feet to the Homes class. These fields will only be displayed for listings using the specified property class.

If you don't wish to utilize the property class system, you can remove all but one property class, and use the single property class for all your listing fields. Open-Realty® will use the single class by default, bypassing the property class selection when searching or creating listings.

- **Class Name** - Displays the name of the property class.
- **Class Rank** - Displays the rank of the property class.
- **Action - Delete** - Click on this to Delete the property class **CAUTION: DELETING A PROPERTY CLASS WILL DELETE ANY/ALL LISTINGS THAT ARE ASSIGNED ONLY TO THAT PROPERTY CLASS** (ie not to any other Property Classes as well)
- **Action - [Modify](#)** - Click on this option to [Modify](#) the property class.
- **[Insert Property Class](#)** - Click on this option to [Insert Property Class](#).

## Insert Property Class

This page is to add a new property class. The available options are:

- **Class Name** - This is the name of the class as it will be stored in the database and displayed to users.
- **Rank** - The rank of the class for the display order of the class. Classes with a lower number will display first.
- **Select the listing fields to add to this class** - This option allows you to select any existing listing fields that you wish to add to this new property class. The insert page is the only place that you will have the ability to mass add listing fields to the property class so it is a good idea to know which fields you want in the property class when adding the property class. Listing fields can still be added later however you must edit each listing field to select the property class.

## Modify Property Class

This page is to modify the Property Class. There is not a whole lot of options available here but here they are.

- **Class Name** - This is the name of the class as it will be stored in the database and displayed to users.
- **Rank** - The rank of the class for the display order of the class. Classes with a lower number will display first.

# Edit Agent/Member Templates

This screen is the Agent and Member Template Editor. They are separate sections however are identical in function so they are being covered in one section of the documentation. The difference will be in the top left hand corner it will say "Member Template Editor" when using that editor instead of the "Agent Template Editor" Here you can edit the fields that are used for your Agent's and Member's accounts.

- [Edit Field](#) - Using the drop down menu select an existing field and select edit to [Edit Field](#)
- [Add Field](#) - Click on this option to [Add a New Field](#)
- [Set Field Order](#) - Click on this option to [Set Field Order](#)

## Add/Edit Agent & Member Fields

The Agent and Member Template Editor is basically the same for Editing or Adding a field. The only difference will be when editing you'll already have information in the fields and will have the **Update & Delete** buttons and when adding you will only have an **Add Field** button.

**Field Name** - This is the name of the field as it will be stored in the database.

**Field Type** - The Type of field that this is. (Text, Textarea, select list, select box, etc....)

**Required** - Is this field required to be filled out?

**Field Caption** - The Caption for the field that will be displayed to users.

**Field Elements** - The choices for multiple selection Field Types such as select lists, option boxes etc. Each choice should be separated by double pipes (||) Do not use spaces or returns after each option, run them all together as one line. For example: Option 1||Option 2||Option 3||Option 4

**Default Text** - The Default text to be displayed in this field

**Field Tool Tip** - Adds a tool tip to the signup and add/edit user pages for the field. (note: requires new css settings. If upgrading or using a custom template, you can copy the a.tooltip... classes from the vertical-menu style.css file.)

**Field Order on Listing Page** - The order that the fields should be displayed in.

**Show Field to** - Use this option to restrict who will see this field. If you have a field you only wish Agents to be able to see you can select "Agents Only" so that only logged in agents will see this field.

## Set Field Order

Here is a single page for setting the field order of all of your agent/member fields at one time. You can edit the **Field Order on Listing Page** value to raise or lower a field's position. Click **Set Order** to save your changes.

# User Manager

The user manager, as the name suggests, allows you to manage your site's users.

The navigation bar at the top of the user manager allows you to enter a specific user ID to edit or sort the site's users by different criteria. From here there is also a button to Add Users.

Below the User Manager area you will have a list of your users displayed and several details for each user.

Clicking on the **Pencil** will allow you to Edit the User. Clicking on the **Red X** will **Delete** the user.

**WARNING!!! ALERT!!! DANGER!!! YIKES!!!**

***Deleting an agent WILL delete ALL of the agent's listings.***

# Editing Users - Edit My Account

The Edit Users (User Manager) the area to the left displays options for editing Images and Files. The area on the right is where you will fill in the user fields that are defined in the Agent or Member Template Editor. Fields with a red \* next to them indicate that field is required and must be completed in order to save the record.

## User Images

The Image Editor is where you can upload, modify or delete images for your user. The image editor is located by first editing a user through the User Manager. You will see on the User Editor thumbnails displayed on the left side of any images already existing for the user and an option to Edit Images. Clicking on the Edit Images option will bring you to the image editor.

**Upload a Picture** - This option will be available if you have less than the maximum allowed images currently uploaded for your user. This value is set in the Site Config. To upload a picture click on browse and select the photo you wish to upload off of your local computer. You will have fields to upload the number of allowed images that you have remaining from the maximum allowed images setting. Click submit and it will be uploaded to your server. See the Image Upload Settings section of the Site Config for more information on uploads.

**Return to Editing Listing** - This will return you to the Modify User page.

**Image Details** - The following fields display details of the uploaded image and thumbnail. It also displays the thumbnail and the full size image.

**Delete** - Clicking this will delete the corresponding image and it's thumbnail.

**Field Order on Listing Page** - This defines the order the image will be displayed. This value is automatically incremented for each new image that is uploaded, you can update this value to change the order the images are displayed.

**Caption** - The caption that will be displayed with the image.

**Description** - The description that will be displayed when viewing the image.

## **Displaying User Images**

To display your user images on your listings, user profile or agent lists you must add a template tag to your user template where you want the images to be displayed. There are several methods for displaying thumbnails of your images and full size images. See the Template Documentation section for available template tags.

## User Files

The File Editor is where you can upload or modify files for your users. The File Editor is located by first editing a user through the User Manager or by clicking on Edit My Account. You will see on the

User Manager a list of existing files on the left side of the user manager and an option to Edit Files. Clicking on the Edit Files option will bring you to the File editor.

**Upload a File** - This option will be available if you have less than the maximum allowed files uploaded for your user. This value is set in the Site Config. To upload a file click on browse and select the file you wish to upload off of your local computer. You will have fields to upload the number of allowed files that you have remaining from the maximum allowed files setting. Click submit and it will be uploaded to your server. See the Upload Settings section of the Site Config for more information on uploads.

**Return to Editing Listing** - This will return you to the User Manager screen.

Below the Upload a File area a list of all the existing files will be displayed which will include an icon of the file type, the file name and the size of the file. Below each file will be the following options:

**Delete** - Clicking this will delete the corresponding image and it's thumbnail.

**Field Order on Listing Page** - This defines the order the file will be displayed. This value is automatically incremented for each new file that is uploaded, you can update this value to change the order the files are displayed.

**Caption** - The caption that will be displayed with the file.

**Description** - The description that will be displayed when viewing the file.

## Displaying Files

To display files on your listings, user profile or agent lists you must add a template tag to your template where you want a list of the available files to appear so they may be downloaded. See the Template Documentation section for available template tags.

# Adding Listings

The first step in adding a new listing is that you will need to select the property class for your listing. By default only one selection is allowed but you can allow multiple selections by changing the option in the Site Configuration.

Once you have selected a property class you will be taken to a page where you will add all the listing data for your listing. Once you have filled out all the required fields and click Submit to save the listing you will be taken to the Modify Listing screen where you can edit your listing's images with the Image Manager, edit Virtual Tours using the Edit Virtual Tour option or add files using the File Manager (See sections in the Editing Listings section of the documentation).

# Editing Listings

The Edit Listings will display all the listings you have available to modify. If you selected "Edit My Listings" you will have only those listings that are owned by you. If you have 'Admin' privileges, or permissions to edit all listings and you selected "Edit All Listings" then you will have a list of all listings available to you.

- **Lookup** - If you already know the listing ID of the listing you want to edit you can enter it in this field and click on Lookup to go directly to editing this listing.
- **Listing Number** - This is the listing ID for each listing.
- **Modify Listing** - Clicking on 'Modify Listing' will take you to the Edit Listing screen where you can edit the details of your listing.
- **Delete Listing** - Clicking on 'Delete Listing' will completely remove the listing and all of it's images from the server.

On the line below these options the 'Listing Title' and any information in the 'Notes' field will be listed for quick reference and to help you identify the listings quickly.

Once you select a listing to modify you will be taken to the Listing Editor. In the Listing Editor the area to the left displays options for editing Listing Images, VTours and Listing Files. The area on the right is where you will fill in the listing fields that are defined in the Listing Template Editor. Fields with a red \* next to them indicate that field is required and must be completed in order to save the listing.

## Listing Images

The Image Editor is where you can upload, modify or delete images for your listings. The image editor is located by first editing a listing through the Listing Editor. You will see on the Listing Editor thumbnails displayed on the left side of any images already existing for the listing and an option to Edit Images. Clicking on the Edit Images option will bring you to the image editor.

**Upload a Picture** - This option will be available if you have less than the maximum allowed images currently uploaded for your listing. This value is set in the Site Config. To upload a picture click on browse and select the photo you wish to upload off of your local computer. You will have fields to upload the number of allowed images that you have remaining from the maximum allowed images setting. Click submit and it will be uploaded to your server. See the Image Upload Settings section of the Site Config for more information on uploads.

**Return to Editing Listing** - This will return you to the Modify Listing page.

**Image Details** - The following fields display details of the uploaded image and thumbnail. It also displays the thumbnail and the full size image.

**Delete** - Clicking this will delete the corresponding image and it's thumbnail.

**Field Order on Listing Page** - This defines the order the image will be displayed. This value is automatically incremented for each new image that is uploaded, you can update this value to change the order the images are displayed.

**Caption** - The caption that will be displayed with the image.

**Description** - The description that will be displayed when viewing the image.

## Displaying Images on Listings

To display your listing images on your listings you must add a template tag to your listing template where you want the images to be displayed. There are methods for displaying thumbnails of your images, full size images, specific images, slideshows and other options. See the Template Documentation section for available template tags.

## **Virtual Tours (VTours)**

The Virtual Tour Editor is where you can upload or modify the Virtual Tours for your listing. For more information about virtual tours read the Virtual Tour Documentation in the Developers Guide section of this documentation.

The VTour editor is located by first editing a listing through the Listing Editor. You will see on the Listing Editor thumbnails displayed on the left side of any VTour images already existing for the listing and an option to Edit VTours. Clicking on the Edit VTours option will bring you to the VTour editor.

**Upload a Picture** - This option will be available if you have less than the maximum allowed Virtual Tours uploaded for your listing. This value is set in the Site Config, or if you are uploading an EGG Solution Virtual Tour then it will only allow 1 per listing. To upload a Virtual Tour click on browse and select the photo you wish to upload off of your local computer. You will have fields to upload the number of allowed images that you have remaining from the maximum allowed images setting. Click submit and it will be uploaded to your server. See the Upload Settings section of the Site Config for more information on uploads.

**Return to Editing Listing** - This will return you to the Modify Listing screen.

**Image Details** - The following fields display details of the uploaded Virtual Tour and thumbnail. It will display the thumbnail and the full size image of the virtual tour or, if you've uploaded an EGG Solution tour it will display a small egg solution icon to indicate you have uploaded that type of Virtual Tour.

**Delete** - Clicking this will delete the corresponding image and it's thumbnail.

**Field Order on Listing Page** - This defines the order the image will be displayed. This value is automatically incremented for each new image that is uploaded, you can update this value to change the order the images are displayed.

**Caption** - The caption that will be displayed with the image.

**Description** - The description that will be displayed when viewing the image.

## Displaying Virtual Tours (VTours) on Listings

To display VTours on your listings you must add a template tag to your listing template where you want a link to your virtual tour to appear. See the Template Documentation section for available template tags.



## **Listing Files**

The File Editor is where you can upload or modify files for your listings. The File Editor is located by first editing a listing through the Listing Editor. You will see on the Listing Editor a list of existing files on the left side of the listing editor and an option to Edit Files. Clicking on the Edit Files option will bring you to the File editor.

**Upload a File** - This option will be available if you have less than the maximum allowed files uploaded for your listing. This value is set in the Site Config. To upload a file click on browse and select the file you wish to upload off of your local computer. You will have fields to upload the number of allowed files that you have remaining from the maximum allowed files setting. Click submit and it will be uploaded to your server. See the Upload Settings section of the Site Config for more information on uploads.

**Return to Editing Listing** - This will return you to the Modify Listing screen.

Below the Upload a File area a list of all the existing files will be displayed which will include an icon of the file type, the file name and the size of the file. Below each file will be the following options:

**Delete** - Clicking this will delete the corresponding file.

**Field Order on Listing Page** - This defines the order the file will be displayed. This value is automatically incremented for each new file that is uploaded, you can update this value to change the order the files are displayed.

**Caption** - The caption that will be displayed with the file.

**Description** - The description that will be displayed when viewing the file.

## **Displaying Files on Listings**

To display files on your listings you must add a template tag to your listing template where you want a list of the available files to appear so they may be downloaded. See the Template Documentation section for available template tags.

# Add-on Manager

The Add-on Manager is a new feature in Open-Realty® for displaying Add-ons, their status and even to remove them.

The Add-on Manager displays a list of all Add-ons installed on your Open-Realty® site. Details of the add-on are as follows:

- **Addon Name** - Displays the name of the Add-on
- **Addon Version** - Displays the Add-on's Version
- **Addon Status** - Displays a status of 'OK' if the add-on is properly installed and up to date. If there is an update available or a problem it will be noted in this column.
- **Actions** - The available Action icons are as follows:
  - **Magnifying Glass** - Check Update - Checks for available updates if supported by the Add-on
  - **Clipboard** - Help - Displays the Add-on Help information if supported by the Add-on. Should display a list of all available Add-on Template Tags and Actions.
  - **Red X** - Uninstall - Uninstalls the add-on, will remove ALL of the Add-on files from the server and, if supported, also remove all database entries created by the Add-on.

# Site Log

The Site Log maintains a list of all actions performed in Open-Realty® in the database for an accountability record. The site log uses a next/prev menu at the top to skip between pages of the log.

The red 'X' is an option to Clear the Log. This option is only available to site admins. Pushing this will ask for a confirmation of the action and then will clear the log, placing a new log entry notating that the log was cleared, when it happened and who cleared the log.

- **ID** - The ID of the logged action
- **Date** - The date and time the logged action occurred
- **User (IP)** - The user ID and the user's IP address are logged
- **Action** - The action that was performed

# TEMPLATE DOCUMENTATION

The template system in Open-Realty® works by replacing {template tags} located in the template files with the real content that they represent. There are multiple template files required by Open-Realty®, all of which have unique template tags available to provide a wide variety of information, data or appearance.

The template system also interfaces with Open-Realty's login system to provide a set of permission tags to allow you to restrict sections of your templates to different user groups.

This section of the documentation contains a list of the different types of templates available in Open-Realty®. Below each of the types of templates, in parenthesis you will see the path (location) of the default template files for those types so you will know where to find them if you wish to edit them or make your own. In the location *TEMPLATE\_NAME* is the name of the template you are using per your configured template in the Site Configuration (Template Settings). In the path for the templates **FILENAME** is listed in place of the specific name of the template file that is used for that template.

Each type of template may have many different versions of that particular template file available. You will see them listed below each type of template along with a description of what they do. Most of the different template types can be selected in the configuration as to which one you want to use on your site while others only use a specific file and others will use different template files based upon the template tag you use to call the function that uses that template (for example the featured listings templates are determined by the template tag used to call that function).

To make your own template from scratch, you will need to maintain the below file naming structure and at the minimum have at least one of the defined template files for each type of template. It may help to use one of the included templates as a guideline or example. Some knowledge of HTML markup will be required in order to create your own template.

# **INSTALLING NEW TEMPLATES**

Additional templates (third party) can be located in the **Custom Template - Releases** sections of the Open-Realty® forums. Templates may also be available from other locations on the internet but the forums are the best place to look first. Once you have located a template that you want to use in Open-Realty® follow the steps below:

- Download the template - Most templates will be distributed in a compressed ZIP file.
- Extract the contents of the compressed file
  - You should be left with a single folder of the template's name that contains all of the template's files. Look in this folder and verify the template files (ie main.html) appear in this folder.
    - In some cases your extracted files may be contained in a second sub folder of the same template name or may be extracted without being contained inside a folder. Make sure the main.html (and the rest of the template files) are in the proper folder: **/templatename/main.html**
- Upload the template folder to your server and place it in the /template/ folder of your site. IE: **yoursite.com/template/templatename**
- Log into the admin area and go to the Site Configuration.
- On the **Template tab** the **Site Template** option drop down list should now display the name of the template folder you just uploaded. Select the new template and save your changes.
- You may also need to select different templates for the Listing Templates, Search Results Templates etc if your new template does not have the same named templates of those types.

## **Upgrading/Updating Templates**

When selecting a template, please make sure it is designed for the OR version that you are using. With the rapid development of OR, template tags are often added or changed and additional features of Open-Realty® are becoming templated and will require an additional template file. If you try to use a template from an older version, it may not function properly due to incorrect template tags or missing template files.

If you receive errors regarding missing template files on your Open-Realty® site then the template you are using was likely designed for an older version and is missing new, required template files. If this happens you can try to correct the problem by comparing your template with one of the included templates. You will need to check for any template files that do not exist in your template and copy them over.

You may also need to update CSS code as new CSS may have been added in order to handle new features of Open-Realty®. If you are unfamiliar with HTML and CSS markup then you may need to seek help on the forums or with a template designer.

Also, please review the template documentation for changes to the template tags and update them as needed.

## **PERMISSION TEMPLATE TAGS**

Each permission tag in the template system are made up of pair of tags that surround the template data that you which to show only to a select group.

For Example:

Opening tags look like this: **{check\_admin}**

Closing tags look like this: **{/check\_admin}**

In addition to the listed template tags you can also add **!** in front of any permission tag to check for NOT having that permission. Where the above example is checking for **admin** permissions, the below examples will check if the user is **NOT** an admin.

For Example:

Opening tags look like this: **{!check\_admin}**

Closing tags look like this: **{/!check\_admin}**

The Following tags are available in all of the templates. You can add the NOT "!" to any of the listed permission tags:

<b>Tag</b>	<b>Description</b>
check_agent	Shows content to agents.
check_guest	Shows content to guest only. Logged in members/agents will not see this info.
check_member	Shows content to members.
check_admin	Shows content to administrators.
check_view_logs	Shows content to administrators or agents with view log permissions
check_edit_pages	Shows content to administrators or agents with edit page permissions
check_have_vtours	Shows content to administrators or agents with virtual tour permissions
check_have_files	Shows content to administrators or agents with file upload permissions
check_edit_site_config	Shows content to administrators or agents with edit site configuration permission.
check_edit_member_template	Shows content to administrators or agents with edit member template permission.
check_edit_agent_template	Shows content to administrators or agents with edit agent template permission.
check_edit_listing_template	Shows content to administrators or agents with edit listing template permission.
check_edit_all_listings	Shows content to administrators or agents with edit all listings permission.
check_edit_all_users	Shows content to administrators or agents with edit all users permission.
check_edit_pages	Shows content to administrators or agents with edit pages permission.

check_edit_listing_classes	Shows content to administrators or agents with edit listing classes permission.
----------------------------	---

# **ADMIN TEMPLATES**

**(/admin/template/TEMPLATE\_NAME)**

These templates control the layout of the administrative area of your Open-Realty® website where agents and admins log in to edit listings, edit the site configuration and perform other tasks. Template tags from the Main Admin Templates will work on any other administrative template page. Other template pages may also have their own set of template specific tags.



# MAIN ADMIN TEMPLATES

Template tags from the Main Admin Templates will work on any other administrative template page.

## Main Admin Template Files:

(/admin/template/TEMPLATE\_NAME/**FILENAME**)

1. **main.html** - This is the template for the main administrative section of the site. It is simply a html file to control the overall design and layout of the entire admin section. The main.html file must have the tag **{content}** to specify where Open-Realty® should place it's information.
2. **or\_index.html** - This is the template used to generate the content area for the index page of the admin section that agents/admins see when they log in.

Each tag in the template system looks like this **{tag\_name}**. The Following tags are available in all Admin Templates.

## Main Admin Template Tags

Tag	Description
license_tag	This places the <b>required</b> html comments into the template. Not including this tag is a violation of the license agreement.
addon_links	This tag parses all of your add-ons, triggers the installation functions and displays the admin icons from the add-on. If you remove this tag, add-on installation functions can not be triggered and add-ons may not work properly.
select_language	Shows the language selection drop down.
version	Shows "Version: 2.X.X". Pulls the currently installed version number and uses the \$lang['version'] variable to show the word version in the correct language.
lang_index_home	Shows the word "Home" from \$lang['index_home'] variable.
lang_index_admin	Shows the word "Admin" from \$lang['index_admin'] variable.
lang_index_logout	Shows the word "Logout" from \$lang['index_logout'] variable.
baseurl	Places the correct baseurl into the template.
template_url	Places the correct admin template url into the template.
load_js_body	This should be the last tag called in the head section of the template. It places all JavaScript required by Open-Realty® into the template as well as the <body> tag and calls any onload functions needed for the page editor. <b>This must be in all main.html admin template files.</b>
load_js	Places JavaScript into the template. This loads javascript defined using the \$load_js variable and can be used throughout Open-Realty® and Add-ons. This should be placed near the top of your main template between the <head> and </head> tags.
load_js_last	Places JavaScript into the template. This loads javascript defined using the \$load_js_last variable and can be used throughout Open-Realty® and Add-ons. This should be placed at the end of your main template just before the </body> tag.
content	This places the actual content generated by Open-Realty® from specific GET['action'] calls.
site_title	This places the site title set in the Open-Realty® site configuration.

company_name	This places the company name from the site configuration onto the page.
company_location	This places the company location from the site configuration onto the page.
company_logo	This places the url for the company logo from the site configuration onto the page.
openrealty_links	Places links to Open-Realty® related resources
general_info	Places a general info block on the admin page providing details about your Open-Realty® installation.
charset	Adds the charset selected in site config to your template.
user_id	Displays the current user ID

# **EDIT LISTINGS TEMPLATE**

This template is used to control the layout of the Edit Listings page which displays a list of the listings that can be edited.

## **Edit Listings Template File:**

*(/admin/template/TEMPLATE\_NAME/FILENAME)*

1. **edit\_listings.html** - This is the Edit Listings template. It has a number of tags available to it, see the edit\_listings\_template\_tags page for details.

Each tag in the template system looks like this **{tag\_name}**. The Following tags are available from the Edit Listing template.

## **Edit Listings Template Tags**

<b>Tag</b>	<b>Description</b>
listing_dataset	BEGINS the section for the display of each listing in the listing editor. This section of the template will be repeated for each listing. This section must be closed with {/listing_dataset}
image_thumb_#	Displays the thumbnail image from the listing where # is the thumbnail number you want displayed.
listingid	Displays the listing ID on the listing view page.
listing_title	This places the listing's title on the page.
listing_pclass	Displays the name of the property class the listing is assigned to. In cases where the listing is part of multiple classes it will show a the names separated by a comma.
listing_field_FIELDNAME	by using the name of a listing field defined in the listing template editor, you can call the individual field and place it on the page. For example to call the "address" field you would insert {listing_field_address} into your template, which would result in both the field caption and value being placed in your template, eg. "Address: 1600 Penn Ave."
listing_field_FIELDNAME_value	Works the same listing_field_FIELDNAME, except you only get the field value. eg. "1600 Penn Ave."
listing_field_FIELDNAME_caption	Works the same listing_field_FIELDNAME, except you only get the field caption. eg. "Address"
listing_field_FIELDNAME_rawvalue	Works the same listing_field_FIELDNAME, except you only get the raw field value. For example when calling the price field instead of getting "\$150,000" you would simply get "150000".
listing_agent_last_name	Displays the listing agent's last name.
listing_agent_first_name	Displays the listing agent's first name.
listing_notes	Displays the contents of the listing's notes field.
listing_active_status	If the listing is active returns "Yes" otherwise returns "No".
listing_featured_status	If the listing is featured returns "Yes" otherwise returns "No".
listing_expiration	Displays the listing's expiration date. (Use with block; show_expiration_block to hide this if expiration is not used)

edit_listing_link	Returns the url to edit the listing for use in a link, example usage: <code>&lt;a href="{edit_listing_link}"&gt;&lt;img src="images/no_lang/listing_editor_edit.jpg" alt="{lang_admin_listings_editor_modify_listing}" width="16" height="16"&gt;&lt;/a&gt;</code>
delete_listing_link	Returns the url to delete the listing for use in a link, example usage: <code>&lt;a href="{delete_listing_link}" onClick="return confirmDelete()"&gt;&lt;img src="images/no_lang/listing_editor_delete.jpg" alt="{lang_admin_listings_editor_delete_listing}" width="16" height="16"&gt;&lt;/a&gt;</code>
email_agent_link	Returns the listing agent's email address
listing_hit_count	Returns the listings hit count
<b>Blocks</b>	
All code contained on the template inside a block will be controlled according to the block it is contained inside. This is for template designers to control the display of many fields or code by enclosing it in one simple template tag block. A STARTING block will look like a normal template tag but that block must be closed with <code>{/TEMPLATE_TAG}</code> The following template tags are blocks:	
show_expiration_block	will hide or show expiration date depending on the use expiration setting
<b>Other/Miscellaneous</b>	
row_num_even_odd	Gets the row number and returns it as 0 or 1. This is used, in combination with the stylesheet to alternate table row colors. Proper usage of this tag in your template would be: <code>&lt;tr class="result_row_{row_num_even_odd}"&gt;</code>

# **PUBLIC SITE TEMPLATES**

***(/template/TEMPLATE\_NAME)***

These templates control the layout of the public portion of your website where visitors to your website will first come and then perform searches, view listings, read various information and contact you or your agents.

Template tags from the Main Templates work on any other public template page. The other template pages may also have their own set of specific tags that will only work on that template file. The Main Templates are the files that control the overall design, layout, text, backgrounds and images displayed as the overall design of your website. Read the description of each of the template files below for more details on what each of them do.

# MAIN TEMPLATES

(/admin/template/TEMPLATE\_NAME/**FILENAME**)

Tags in these templates will work in any of the public templates as well as when placed in the WYSIWYG page editor generated pages.

1. **main.html** - This is the template for the main site. It is equivalent to the user\_top.html and user\_bottom.html files for Open-Realty® 1.x. This template file is **THE** design for your website. All of the design/layout of your site is controlled with this file. The background, header images, CSS styling, static content etc is all defined in this file. It is simply a html file with the tag **{content}** to specify where Open-Realty® should place it's information. When viewing your site through Open-Realty® the **{content}** tag is replaced with information defined by the sub template for the action you are trying to perform. For example when viewing listings, the content area will be replaced with content using the listing\_detail\_default.html file, or from other template files or direct content from Open-Realty® depending on which page of the site you are viewing.

2. **popup.html** - This is the template used on popup windows such as the Calculators and Agent Contact Forms. The template contains head/body/html tags but is otherwise a blank template. Use this template by setting &popup in the URL.

3. **blank.html** - This is a completely blank and empty template file used for popup windows by setting &popup=blank

4. **printer\_friendly.html** - This is the template used when the user selects the Printer Friendly Page from listing details.

## **Optional**

5. **page#\_main.html** - This is an optional main template for the site that is based on the page number. If you wish to have a unique template design for a specific page such as having featured listings in a section of the template on the home page (Page 1) but not the rest of the site then you can create a page#\_main.html template with the differences you want.

Open-Realty® will use the page#\_main.html template for pages where the alternate main template exists, if there is no specific main template for that page then it will default to the main.html template for that page. (*# would be the page number you want to use this main template for*)

You can specify a PageID=# variable in the URL for any page on your site in order to specify an alternate main template file for the page you are displaying. The PageID=# variable is specified automatically for all page editor created pages, you can create a custom PageID for any other page you wish to have an alternate main template layout for.

6. **searchresults\_main.html** - This is an optional main template that allows you to load a unique template for your search results page. This can be used to define custom metatags for your search result pages or to simply give your search results a different look from the main template.

7. **maintenance\_mode.html** - This template is used and displayed to site visitors when the site is in "Maintenance Mode". (See the Maintenance Mode option in the Site Configuration General tab)

8. **500.shtml** - This is not a template that is used inside Open-Realty®. This template is a 500 Internal Server error page that will be displayed to the user if there is a database error instead of displaying the SQL error on the page. This is for security and aesthetic purposes.

## Main Template Tags

Each tag in the template system looks like this **{tag\_name}**. The Following tags are available from the main templates.

Tag	Description
license_tag	This places the <b>REQUIRED</b> html comments into the template. <b><u>Not including this tag is a violation of the license agreement.</u></b> If you would like to remove this license tag you must purchase a Commercial License for Open-Realty®.
addthis_button	Places an "AddThis" button on your site where the template tag is specified. AddThis supports 150+ services such as Digg, Stumbleupon, Facebook, MySpace, Twitter and 100's of other services. This loads external javascript from the AddThis site to function.
select_language	Shows the language selection dropdown. This is <b>NOT</b> yet supported.
templated_search_form	This places the templated search form onto your main template page or into any of the WYSIWYG pages.
baseurl	Places the correct baseurl into the template.
url_index	Places the URL for the Index Page into the template.
url_search	Places the URL to the Search Page into the template.
url_search_rental	Places the URL to the Rental Search page into the template.
url_search_class_#	Places the URL to the search page for property class # into the template.
url_search_results	Places the URL to the Search Results page into the template. This will be to display ALL listings
url_searchresults_class_#	Places the URL to the search results page for property class # into the template.
url_view_agents	Places the URL to view the agent's into the template.
url_view_calculator	Places the URL to view the calculator into the template.
url_view_favorites	Places the URL to View a member's Favorite listings. This should be used inside of the check member permission tags.
url_view_saved_searches	Places the URL to view a member's Saved Searches. This should be used inside of the check member permission tags.
url_logout	Places the URL for a member to LOGOUT of the site. This should be used inside of the check member permission tags.
url_member_signup	Places the URL for a visitor to sign up as a member to the site. This should be used inside of the check guest permission tags.
url_member_login	Places the URL for a visitor to LOGIN to the site as a member. This should be used inside of the check guest permission tags.
url_agent_signup	Places the URL for a visitor to sign up as an agent on the site. This should be used inside of the check guest permission tags.

url_agent_login	Places the URL for an agent to LOGIN to the site as an agent. This should be used inside of the check guest permission tags.
url_blog	Places the URL for the Blog Index on the site. The blog index displays a list of all the blog posts
page_link_#	Places a link to the specified page # into the template. The page number is provided when editing a page with the Page Editor.
blog_link_#	Places a link to the specified blog post # into the template. The blog post # is provided when editing a blog post in the Blog Manager.
featured_listings_vertical	Inserts the featured listings in a vertical layout.
featured_listings_horizontal	Inserts the featured listings in a horizontal layout.
random_listings_vertical	Inserts random listings in a vertical layout.
random_listings_horizontal	Inserts random listings in a horizontal layout.
latest_listings_vertical	Inserts the latest listings in a vertical layout.
latest_listings_horizontal	Inserts the latest listings in a horizontal layout.
featured_listings_vertical_class_#	Inserts the featured listings for property class # in a vertical layout.
featured_listings_horizontal_class_#	Inserts the featured listings for property class # in a horizontal layout.
random_listings_vertical_class_#	Inserts random listings for property class # in a vertical layout.
random_listings_horizontal_class_#	Inserts random listings for property class # in a horizontal layout.
latest_listings_vertical_class_#	Inserts the latest listings for property class # in a vertical layout.
latest_listings_horizontal_class_#	Inserts the latest listings for property class # in a horizontal layout.
load_js	Places JavaScript into the template. This loads javascript defined using the \$load_js variable and can be used throughout Open-Realty® and Add-ons. This should be placed near the top of your main template between the <head> and </head> tags.
load_js_last	Places JavaScript into the template. This loads javascript defined using the \$load_js_last variable and can be used throughout Open-Realty® and Add-ons. This should be placed at the end of your main template just before the </body> tag.
load_meta_keywords	This places the meta keywords tag and values into your template. The Meta keywords help with search engine ranking. This tag should be placed inside the head tag of your main.html template.
load_meta_description	This places the meta description tag and values into your template. The Meta description helps with search engine ranking. This tag should be placed inside the head tag of your main.html template.
load_meta_keywords_raw	This places the raw meta keywords values into your template. The Meta keywords help with search engine ranking. This tag should be placed inside the head tag of



	your main.html template. Proper usage of this tag would be within the meta keywords tag for example: <code>&lt;meta name="keywords" content="{load_meta_keywords_raw}" /&gt;</code>
load_meta_description_raw	This places the raw meta description values into your template. The Meta description helps with search engine ranking. This tag should be placed inside the head tag of your main.html template. Proper usage of this tag would be within the meta description tag for example: <code>&lt;meta name="description" content="{load_meta_description_raw}" /&gt;</code>
template_url	Places the correct template url into the template.
content	This Places the actual content generated by Open-Realty® from specific GET['action'] calls.
site_title	This places the site title set in the Open-Realty® site configuration.
company_name	This places the company name from the site configuration onto the page.
company_location	This places the company location from the site configuration onto the page.
company_logo	This places the url for the company logo from the site configuration onto the page.
lang_VARIABLENAME	This places the specified language variable from your language file into the template. For example using {lang_featured_listings} would call the language variable: <b>\$lang['featured_listings']</b> and would display its value on the template, in this case: <b>Featured Listings</b>
load_css_FILENAME	Loads the specified CSS file into your template. The CSS file must be located in your template directory with main.html and have a .css extension. This tag should be placed inside your html head tags.
charset	Adds the charset selected in site config to your template.
template_select	Shows the template selection dropdown box.

# **LISTING DETAIL TEMPLATES**

*(/template/TEMPLATE\_NAME/FILENAME)*

These Templates are used when the user selects to view a listing. Which Template is used is determined by the administrator from the configuration screen. Any template named listing\_detail\_templatename.html will be shown as an option in the site configuration.

1. **listing\_detail\_default.html** - This is the default listing View template. It has a number of tags available to it, see the listing\_template\_tags page for details.
2. **listing\_detail\_mainjavacolumn.html** - This is the Listing View with Java Image with thumbnails in columns.
3. **listing\_detail\_slideshow.html** - This is the Listing View with Slideshow.
4. **listing\_detail\_mainjavarows.html** - This is the Listing View with Java Image with thumbnails in rows.
5. **listing\_detail\_tabbed.html** - This is the Listing View template with all the elements layed out using Tabs.

## **Optional**

6. **listing\_detail\_pclass#.html** - This is an optional listing detail template based on the property class where the # is the property class ID. If you create a listing detail template with this naming for a specific property class then it will be used for listings in that property class, overriding your listing detail setting in the site config. When a template doesn't exist for a specific property class the setting in the site config will be used.

For example, if you have:

Residential = 1

Land = 2

Commercial = 3

Multifamily= 4

If you want different listing details pages for each class above your templates would need to be named as follows:

**listing\_detail\_pclass1.html** (Residential)

**listing\_detail\_pclass2.html** (Land)

**listing\_detail\_pclass3.html** (Commercial)

**listing\_detail\_pclass4.html** (Multifamily)

## Listing Detail Template Tags

Each tag in the template system looks like this **{tag\_name}**. The Following tags are available from the listing detail templates.

Tag	Description
listingid	Displays the listing ID on the listing view page.
headline	Displays all listing fields set to show in the headline area from the listing template editor.
featured_listings_vertical	Inserts the featured listings in a vertical layout.
featured_listings_horizontal	Inserts the featured listings in a horizontal layout.
vtour_button	Places a link to display the listings virtual tours. Link will only display if there is a virtual tour.
listing_images	Places the listing images on the page.
listing_images_nocaption	Places the listing images on the page without the image captions.
top_left	Displays all listing fields set to show in the top left area from the listing template editor.
top_right	Displays all listing fields set to show in the top right area from the listing template editor.
center	Displays all listing fields set to show in the center area from the listing template editor.
feature1	Displays all listing fields set to show in the feature1 area from the listing template editor.
feature2	Displays all listing fields set to show in the feature2 area from the listing template editor.
bottom_left	Displays all listing fields set to show in the bottom left area from the listing template editor.
bottom_right	Displays all listing fields set to show in the bottom right area the listing template editor.
contact_agent_link	This places a link to "Contact Agent". The link will open a popup window with the contact agent form.
contact_agent_link_url	This provides the raw html address for the "Contact Agent". This can be used to create your own link to the Contact agent page using Html.
link_calc	This places a link to the calculators.
link_calc_url	This provides the raw html address for the calculators. This can be used to create your own link to the calculators using HTML. For example: <code>&lt;a href="{link_calc_url}"&gt;Calculator Link&lt;/a&gt;</code> (The raw html address does not contain ANY popup, size or other code found in the normal link tag)
link_add_favorites	This places a link to allow members to add the listing to there favorites list.
link_add_favorites_url	This provides the raw html address for the add to favorites. This can be used to create your own link to add to favorites using HTML. For example: <code>&lt;a href="{link_add_favorites_url}"&gt;Add Favorites</code>

	<i>Link</i>
link_printer_friendly	This places a link to show the listing in the printer friendly template.
link_printer_friendly_url	This provides the raw html address for the printer friendly page. This can be used to create your own link to the printer friendly page using HTML. For example: <i>&lt;a href="{link_printer_friendly_url}"&gt;Printer Friendly Link&lt;/a&gt;</i> (The raw html address does not contain ANY popup, size or other code found in the normal link tag)
link_email_friend	This places a link to allow visitors to email their friends about the listing.
link_email_friend_url	This provides the raw html address for the email friend form. This can be used to create your own link to the email friend form using HTML. For example: <i>&lt;a href="{link_email_friend_url}"&gt;Email Friend Link&lt;/a&gt;</i> (The raw html address does not contain ANY popup, size or other code found in the normal link tag)
link_map	This places a link to the map system defined in the site configuration.
link_map_url	This provides the raw html address for the map link. This can be used to create your own link to the map using HTML. For example: <i>&lt;a href="{link_map_url}"&gt;View Map&lt;/a&gt;</i> (The raw html address does not contain ANY popup, size or other code found in the normal link tag)
link_yahoo_school	This places a link to the yahoo school information for the listing area.
link_yahoo_school_url	This provides the raw html address for the yahoo school information link. This can be used to create your own link to the yahoo school information using HTML. For example: <i>&lt;a href="{link_yahoo_school_url}"&gt;View School Information&lt;/a&gt;</i> (The raw html address does not contain ANY popup, size or other code found in the normal link tag)
link_yahoo_neighborhood	This places a link to the yahoo neighborhood information for the listing area.
link_yahoo_neighborhood_url	This provides the raw html address for the yahoo neighborhood information link. This can be used to create your own link to the yahoo neighborhood information using HTML. For example: <i>&lt;a href="{link_yahoo_neighborhood_url}"&gt;View Neighborhood Information&lt;/a&gt;</i> (The raw html address does not contain ANY popup, size or other code found in the normal link tag)
link_edit_listing	This places a link to edit the listing on the listing detail page if the agent is logged in and is the owner of the listing, or if the logged in agent is an admin or has permissions to edit all listings.
link_edit_listing_url	This provides the raw html address to the edit listing page on the listing detail page if the agent is logged in

	and is the owner of the listing, or if the logged in agent is an admin or has permissions to edit all listings.
listing_agent_name	This places the listing agent's name.
listing_agent_link	This places a link to the listing agent's information
listing_agent_listings	The places a link to the search results to view other listings from the listing agent.
listing_agent_id	The places the listing agents id on the page.
listing_title	This places the listing's title on the page.
hitcount	This places the number of time the listing has been viewed.
slideshow_images	Places Images into a slideshow.
main_image	Places a large main listing image on the page.
main_image_nodesc	Places a large main listing image on the page. This version of the tag does not display the image description below the main image.
image_thumb_#	Displays a listing's thumbnail image. The number corresponds to the to the image order defined in the listing image editor.
raw_image_thumb_#	Same as image_thumb_# except it returns the image url to be used in an image tag.
main_image_java	Places the main listing image on the page for the java image switch.
main_image_java_nodesc	Places the main listing image on the page for the java image switch. This version of the tag does not display the image description below the main image.
listing_images_java	Places the listing images in a vertical column, when clicked, the image will replace the current main image on the page. Requires main_image_java also be on page.
listing_images_java_caption	Places the listing images in a vertical column with the image caption below each image, when clicked, the image will replace the current main image on the page. Requires main_image_java also be on page.
listing_images_java_rows	Places the listing images in a row, when clicked it the image will replace the current main image on the page. Requires main_image_java also be on page.
listing_images_mouseover_java	Places the listing images in a vertical column, when the mouse is over the image, the image will replace the current main image on the page. Requires main_image_java also be on page.
listing_images_mouseover_java_caption	Places the listing images in a vertical column with the image caption below each image, when the mouse is over the image, the image will replace the current main image on the page. Requires main_image_java also be on page.
listing_images_mouseover_java_rows	Places the listing images in a row, when the mouse is over the image it the image will replace the current main image on the page. Requires main_image_java also be on page.

image_full_X	Displays the specified main/full image where X = the number of the order of the main image specified in the image editor.
raw_image_full_X	Displays the specified main/full image NAME ONLY, where X = the number of the order of the main image specified in the image editor.
image_full_fullurl_X	Displays the specified main/full image at the configured display size. The image is linked to the image viewing path using the full URL. Where X = the number of the order of the main image specified in the image editor.
listing_field_FIELDNAME	by using the name of a listing field defined in the listing template editor, you can call the individual field and place it on the page. For example to call the "address" field you would insert {listing_field_address} into your template, which would result in both the field caption and value being placed in your template, eg. "Address: 1600 Penn Ave."
listing_field_FIELDNAME_value	Works the same listing_field_FIELDNAME, except you only get the field value. eg. "1600 Penn Ave."
listing_field_FIELDNAME_caption	Works the same listing_field_FIELDNAME, except you only get the field caption. eg. "Address"
listing_field_FIELDNAME_rawvalue	Works the same listing_field_FIELDNAME, except you only get the raw field value. For example when calling the price field instead of getting "\$150,000" you would simply get "150000".
next_prev	Displays the next - prev options on the Listing Detail page. To use you will need to place the tag on the listing detail template AND turn on "Show NextPrev Bar On Listing Page" in Configuration_listing_settings .  <b>WARNING:</b> This will cause a performance hit on sites with large databases that might yield large search results.
next_prev_bottom	Displays the next - prev options on the Listing Detail page. To use you will need to place the tag on the listing detail template AND turn on "Show NextPrev Bar On Listing Page" in Configuration_listing_settings. This uses the next_prev_bottom.html template.  <b>WARNING:</b> This will cause a performance hit on sites with large databases that might yield large search results.
listing_pclass	Displays the name of the property class the listing is assigned to. In cases where the listing is part of multiple classes it will show a the names separated by a comma.
listing_agent_last_name	Displays the listing agent's last name.
listing_agent_first_name	Displays the listing agent's first name.
listing_agent_thumbnail_#	Displays the listing agent's thumbnail image that is specified in the # place. Whatever number is specified will display the thumbnail that falls in that order. <i>note:</i> first image is <b>0</b> , NOT 1

listing_agent_field_FIELDNAME	by using the name of an agent field defined in the user template editor, you can call the individual field and place it on the page. For example to call the "phone" field you would insert {listing_agent_field_phone} into your template, which would result in both the field caption and value being placed in your template, eg. "Phone: 555-1234"
listing_agent_field_FIELDNAME_value	Works the same listing_agent_field_FIELDNAME, except you only get the field value. eg. "555-1234"
listing_agent_field_FIELDNAME_rawvalue	Works the same listing_agent_field_FIELDNAME, except you only get the raw, unformatted field value. For example when calling the price field instead of getting "\$150,000" you would simply get "150000".
listing_agent_field_FIELDNAME_caption	Works the same listing_agent_field_FIELDNAME, except you only get the field caption. eg. "Phone"
show_vtour	Embeds the VTour on your listing page. This will use the selected VTour template to display the VTour in your listing page. You should use the vtour_tab_block template block around this tag to hide the vtour if there is not a supported vtour uploaded for that listing.
fulllink_to_listing	absolute link to the listing instead of the relative link, for use with the RSS Newsfeeds.
get_creation_date	Displays the listing's creation date in the configured Open-Realty® date format.
get_modified_date	Displays the listing's last modification date in the configured Open-Realty® date format.
listing_files_select	Places a simple drop down file selection list onto the template for users to select the file they wish to download.
files_listing_vertical	Places the vertical templated listing files list on the template for users to view and select the files they wish to download.
files_listing_horizontal	Places the horizontal templated listing files list on the template for users to view and select the files they wish to download.
get_featured	Adds "featured" to listing details or search results if listing is featured otherwise does nothing. Useful for CSS templating purposes or just letting people know this is a featured listing.
get_featured_raw	Adds "yes" or "no" to the template depending on if the listing is featured or not. Primarily used in CSS for templating purposes.
money_sign	Adds the configured money sign to the template.
<b>Blocks</b>	
All code contained on the template inside a block will be controlled according to the block it is contained inside. This is for template designers to control the display of many fields or code by enclosing it in one simple template tag block. A STARTING block will look like a normal template tag but that block must be closed with {/TEMPLATE_TAG} The following template tags are blocks:	
hide_printer_friendly	Hides the content contained inside this block when the printer friendly page is being displayed. You must close this block with /hide_printer_friendly

show_printer_friendly	Shows the content contained inside this block when the printer friendly page is being displayed. You must close this block with /show_printer_friendly
show_listed_by_admin_block	Creates a block for listed by admin data. Uses the "Show Listedby Link For Admin's Listings" setting in the Configuration_listing_settings . You must close this block with /show_listed_by_admin_block
!show_listed_by_admin_block	Creates a block to be displayed if this is an admin listing and if we have the Show Listedby Admin option disabled. Uses the "Show Listedby Link For Admin's Listings" setting in the Configuration_listing_settings . You must close this block with /!show_listed_by_admin
vtour_tab_block	Hides the content contained inside this block if there is no supported vtour uploaded for a listing. This tag is primarily for use on the tabbed template but can be used around the show_vtour tag if embedding a vtour on your listing page. You must close this block with /vtour_tab_block
listing_field_FIELDNAME_block	To be placed around listing_field_FIELDNAME tags to prevent display of any code contained inside this block into the template if the field is not to be shown. You must close this block with /listing_field_FIELDNAME_block
listing_agent_field_FIELDNAME_block	To be placed around listing_agent_field_FIELDNAME tags to prevent display of any code contained inside this block into the template if the field is not to be shown. You must close this block with /listing_agent_field_FIELDNAME_block



# **SEARCH PAGE TEMPLATES**

*(/template/TEMPLATE\_NAME/FILENAME)*

This template is used to control the layout of the search page.

Most of this template is created dynamically by the software base upon which fields you have specified as searchable. Due to this there is only one search page template and is NOT selectable in the configuration.

1. **search\_page\_default.html** - This is the default search result template.

## **Optional**

2. **search\_page\_class\_CLASSID.html** - IF there is a template file with this naming convention where the CLASSID is the number of the property class being searched then this search page template will be used instead of the default search page.

The Search Page fields must be contained inside a form. Use the following code for your template form:

```
<form action="index.php" method="get">  
CONTENT  
<input type="hidden" name="action" value="searchresults" />  
<input type="submit" />  
</form>
```

Replace the "CONTENT" above with any formatting code and template tags you wish to use for your template. You can look at the source code for the search\_page\_default.html file as an example to assist you.

## **Custom Search Pages**

It is possible to forgo using Open-Realty's somewhat limited Search Page Template system and create your own custom search forms. You will still use the initial code on the template for the form content but replace the CONTENT area with your own custom form fields and their values.

If you are unfamiliar with how to do this then I suggest you use Open-Realty's existing search form template

You can pass the following options to the search now. Value can be a Unix integer timestamp or an ISO format Y-m-d H:i:s

**Find all listings modified last before Feb 12 2006 at midnight**  
***&listing\_last\_modified\_less=2006-2-12***

**Find all listings modified last on Feb 12 2006 at 7:30:00 PM**  
***&listing\_last\_modified\_equal=2006-2-12%2019:30:00***

**Find all listings modified last after Feb 12 2006 at 8:00:00 PM**  
***&listing\_last\_modified\_greater=2006-2-12%2020:00:00***

## Multiple Sortby Values

Some people may want to sort listings by multiple fields. Open-Realty® now supports sorting by up to 2 fields, separated by a comma (see examples below).

For example you may want to have your listings sorted by City so all listings are grouped by their city and then sorted by price within those groups. Placing the city as the first sortby and the price as the second sortby will accomplish this. You will need to add a hidden field to your search form to add the additional sortbys to pass it to the search results string.

This example will sort listings by Featured Listings first and then by price. So all of your listings that are featured will appear first and then the rest of the listings will appear. They will all be sorted by price as well.

***index.php?action=searchresults&sortby=listingsdb\_featured,price&sorttype=DESC,ASC***

This example will sort the listings by City first and then price. All of your listings will be grouped by city and then sorted by price within those groupings.

***index.php?action=searchresults&sortby=city,price&sorttype=DESC,ASC***

## Additional Search String Variables

In addition to the other documented search strings, and the ones generated by the search form based on selected criteria you can also pass the following strings to the search results URL for specific results:

**&featuredOnly=yes** = This will give results that will display only featured listings.

## Search Page Template Tags

Each tag in the template system looks like this **{tag\_name}**. The Following tags are available from the search page templates.

Tag	Description
featured_listings_NAME	Renders featured listings using the featured listings template specified in place of "NAME"
browse_all_listings	Renders a text link to "Browse all listings" and shows the total number of listings
search_fields	This is the MOST IMPORTANT field. This displays all of the searchable listing fields. This MUST be inside the <form> </form> tags.
agent_searchbox	Displays the drop down agent search box.
show_only_with_images	Displays the Form check box for "Show only listings with images" on the search page.
lang_variable	Displays the value for any defined lang variable. Examples: {lang_show_only_with_images} {lang_show_only_with_vtours} {lang_search_listings}
show_only_with_vtours	Displays the Form check box for "Show only listings with virtual tours" on the search page.
search_type	Adds a hidden field to the search form that includes the property class for the search. This is part of the new property class system of Open-Realty® 2.1.
postalcode_dist_search	Adds a distance search option to the search page using zip codes. <i>The distance search functions require the use of additional data not provided by Open-Realty®. See the Distance Search Database for details and instructions.</i>
lat_long_dist_search	Adds a distance search option to the search page using Latitude and Longitude positions. Your listings must have listing fields of <b>latitude</b> and <b>longitude</b> and have these values filled in already to use this feature (such as data provided by several MLS systems). Open-Realty® will not Geocode an address automatically. <i>The distance search functions require the use of additional data not provided by Open-Realty®. See the Distance Search Database for details and instructions.</i>
browse_all_listings_pclass	Shows you the browse all listing link, except it links to the listings in the selected property class and shows a count of listings in the current property class. If no property class is selected it falls back to the old display_all_listing link.
full_text_search	Adds a search box that will compare the users entry to all listing field values.
city_dist_search	Adds a distance search option to the search page using city names. <i>The distance search functions require the use of additional data not provided by Open-Realty®. See the Distance Search Database for details and instructions.</i> Note: This will not be 100% accurate as some city names have multiple zip codes and some are located in more than 1 state.

### Blocks

All code contained on the template inside a block will be controlled according to the block it is contained inside. This is for template designers to control the display of many fields or code by

enclosing it in one simple template tag block. A STARTING block will look like a normal template tag but that block must be closed with `{/TEMPLATE_TAG}` The following template tags are blocks:

templated_search_form	The search form should be contained inside this block. When a call is made for the templated search form the code inside this block will be what is displayed. This is used for including the search form on pages other than the search page.
no_search_results_block	The code inside this block will be displayed when a search is performed and there are no results. This will be in addition to the code on the rest of the template.

# **SEARCH RESULTS TEMPLATES**

*(/template/TEMPLATE\_NAME/FILENAME)*

This template is used to control the layout of the search result page, this is the most complex template file, as the search results are dependent upon the listing field setup/configuration. Changes to the listing fields can effect the layout and data that appears on the search results templates.

Which Search Results Template that is used is determined by the site configuration. Any template named search\_result\_templatename.html will be shown as an option in the site configuration.

1. **search\_result\_default.html** - This is the default search result template. It has a number of tags available to it, see the search\_result\_template\_tags page for details.
2. **search\_result\_box.html** - This is a tabled layout for the search results containing more fields and information in a nicely formatted table.
3. **search\_result\_nophoto.html** - This is a simple single line of listing information without any thumbnail photo displayed.
4. **search\_result\_tabbed.html** - This is a basic implementation of the tabbed javascript code into a search results template to show the basic design and requirements of using the tabbed code on the search results.

## **Optional**

5. **search\_results\_class\_CLASSID.html** - IF there is a template file with this naming convention where the CLASSID is the number of the property class being searched then this search results template will be used, overriding your configuration's selected search results template.

For example, if you have:

Residential = 1  
Land = 2  
Commercial = 3  
Multifamily= 4

If you want different search results templates for each class above your templates would need to be named as follows:

**search\_results\_class\_1.html** (Residential)  
**search\_results\_class\_2.html** (Land)  
**search\_results\_class\_3.html** (Commercial)  
**search\_results\_class\_4.html** (Multifamily)

## Search Results Template Tags

Each tag in the template system looks like this **{tag\_name}**.

The **#** for the thumbnails this is the order the thumbnails appear, so 1 would display the first thumbnail image, 2 the second etc.

The **#** for the fields, headers and textareas is equal to the **"Field Order on Search Result Page"** setting in the Listing Template Editor.

Keep in mind that the fields to be displayed on the Search Result template must also have **"Display On Search Result Page"** set to **"Yes"** or the field will not be displayed.

The Following tags are available from the search results templates.

Tag	Description
next_prev	Inserts the Next_Prev function into the search results page. This uses the next_prev.html template
next_prev_bottom	Inserts the Next_Prev function into the search results page. This uses the next_prev_bottom.html template
header_title	Displays the title field name "Title" in the header area. This will be linked as a sortable field.
header_title_no_sort	Same as above without link. Text Only
header_#	Displays the Field name in the header area. This will be linked as a sortable field. The number is defined in the "Sort Order" option in the listing template editor.
header_#_no_sort	Same as above without link. Text Only
header_pclass	Displays "Property Class" in the header area. This is linked as a sortable field. This will sort the search results by property class
header_pclass_no_sort	Same as above without link. Text Only.
field_title	Displays the Title field value.
image_thumb_#	Displays a listing's thumbnail image. The number corresponds to the to the image order defined in the listing image editor.
raw_image_thumb_#	Same as image_thumb_# except it returns the image url to be used in an image tag.
image_full_X	Displays the specified main/full image where X = the number of the order of the main image specified in the image editor.
raw_image_full_X	Displays the specified main/full image NAME ONLY, where X = the number of the order of the main image specified in the image editor.
image_full_fullurl_X	Displays the specified main/full image at the configured display size. The image is linked to the image viewing path using the full URL. Where X = the number of the order of the main image specified in the image editor.
field_#	Displays the Data from the specified field number. The number is defined in the "Sort Order" option in the listing template editor.

textarea_#	Displays entire field value from a Text Area
textarea_#_short	Displays first 100 characters from a Text Area then displays "... More Info" which is linked to the listing view.
delete_from_favorite	Displays the option to delete a listing from a user's saved favorites. Option will only be displayed when in favorites.
vtour_button	Displays the Virtual Tour link button on the search results for each listing that has a virtual tour. NOTE: in Open-Realty® 2.5 and earlier this tag is called with {vtour_link} and was changed in order to be identical to the tag that is called from the listing detail page.
link_to_listing	Allows you to create a link to the listing on the Search Results page. For Example: <a href="{link_to_listing}">Click here for listing details</a>
listing_pclass	Displays the name of the property class the listing is assigned to. In cases where the listing is part of multiple classes it will show a the names separated by a comma.
listingid	Displays the Listing ID on the search results page.
listing_field_FIELDNAME	by using the name of a listing field defined in the listing template editor, you can call the individual field and place it on the page. For example to call the "address" field you would insert {listing_field_address} into your template, which would result in both the field caption and value being placed in your template, eg. "Address: 1600 Penn Ave."
listing_field_FIELDNAME_value	Works the same listing_field_FIELDNAME, except you only get the field value. eg. "1600 Penn Ave."
listing_field_FIELDNAME_caption	Works the same listing_field_FIELDNAME, except you only get the field caption. eg. "Address"
listing_field_FIELDNAME_rawvalue	Works the same listing_field_FIELDNAME, except you only get the raw field value. For example when calling the price field instead of getting "\$150,000" you would simply get "150000".
listing_agent_last_name	Displays the listing agent's last name.
listing_agent_first_name	Displays the listing agent's first name.
listing_agent_thumbnail_#	Displays the listing agent's thumbnail image that is specified in the # place. Whatever number is specified will display the thumbnail that falls in that order
listing_agent_field_FIELDNAME	by using the name of a user field defined in the user template editor, you can call the individual field and place it on the page. For example to call the "phone" field you would insert {listing_agent_field_phone} into your template, which would result in both the field caption and value being placed in your template, eg. "Phone: 555-1234"
listing_agent_field_FIELDNAME_value	Works the same listing_agent_field_FIELDNAME, except you only get the field value. eg. "555-1234"
listing_agent_field_FIELDNAME_rawvalue	Works the same listing_agent_field_FIELDNAME, except you only get the raw, unformatted field value. For

	example when calling the price field instead of getting "\$150,000" you would simply get "150000".
listing_agent_field_FIELDNAME_caption	Works the same listing_agent_field_FIELDNAME, except you only get the field caption. eg. "Phone"
fulllink_to_listing	The absolute link to the listing instead of the relative. Used for RSS Newsfeeds.
link_edit_listing	This places a link to edit the listing on the search results page for each listing if the agent is logged in and is the owner of the listing, or if the logged in agent is an admin or has permissions to edit all listings.
link_edit_listing_url	This provides the raw html address to the edit listing page on the search results page for each listing if the agent is logged in and is the owner of the listing, or if the logged in agent is an admin or has permissions to edit all listings.
get_creation_date	Displays the listing's creation date in the configured Open-Realty® date format.
get_modified_date	Displays the listing's creation date in the configured Open-Realty® date format.
get_featured	Adds "featured" to listing details or search results if listing is featured otherwise does nothing. Useful for CSS templating purposes or just letting people know this is a featured listing.
get_featured_raw	Adds "yes" or "no" to the template depending on if the listing is featured or not. Primarily used in CSS for templating purposes.
money_sign	Adds the configured money sign to the template.
isfavorite	Returns "yes" if listing is marked as a favorite listing by a logged in user and "no" if not marked as favorite or the user is not logged in. Can be used for CSS etc.
link_add_favorites	Displays a link to add the listing to the user's saved favorite listings.
link_add_favorites_url	This provides the raw URL to the add to favorites page so that it can be placed in a link or have an image linked by the template designer.

### **Blocks**

All code contained on the template inside a block will be controlled according to the block it is contained inside. This is for template designers to control the display of many fields or code by enclosing it in one simple template tag block. A STARTING block will look like a normal template tag but that block must be closed with `{/TEMPLATE_TAG}` The following template tags are blocks:

<i>TEMPLATE_TAG_block</i>	BEGINS a block for the defined template tag. All code contained on the template inside this block will not be displayed if there is no matching field. This is for template designers who do not know the number of fields the site admin will have marked as searchable. It does not hide the data if the field exist but is empty. Block must be closed with <code>{/TEMPLATE_TAG_block}</code> The following template tags have block options: header_title; header_title_no_sort; header_#; header_#_no_sort; field_title; field_#; textarea_#; textarea_#_short
---------------------------	---



search_result_header	<p>BEGINS the Search Result header area. This area will only be displayed once on the page.  This section must be closed with  {/search_result_header}</p>
search_result_dataset	<p>BEGINS the section for the display of the search results. This section of the template will be repeated for each search result.  This section must be closed with  {/search_result_dataset}</p>
<b>Other/Miscellaneous</b>	
row_num_even_odd	<p>Gets the row number and returns it as 0 or 1. This is used, in combination with the stylesheet to alternate table row colors. Proper usage of this tag in your template would be: <b>&lt;tr class="result_row_{row_num_even_odd}"&gt;</b></p>

# **FEATURED/RANDOM LISTINGS TEMPLATES**

*(/template/TEMPLATE\_NAME/FILENAME)*

**The below information applies to both the featured and random listing options.**

This template is used to control the layout of the featured listings.

Which Template is used is determined by the featured or random listing template tag used on your site. The Featured Listings template tag is: {featured\_listings\_NAME} where NAME is the name of the featured listings template included in your template directory.

For example using the included templates: Using {featured\_listings\_vertical} will use featured\_listing\_vertical.html while using {featured\_listings\_horizontal} will use featured\_listing\_horizontal.html.

1. **featured\_listing\_vertical.html** - This is the vertical featured listings template.
2. **featured\_listing\_horizontal.html** - This is the horizontal featured listings template.

## Featured/Random Template Tags

Each tag in the template system looks like this **{tag\_name}**.  
 The Following tags are available from the featured/random listings templates.

Tag	Description
lang_variable	Displays the value for any defined lang variable.
listingid	Displays the listing ID for the listing being displayed.
featured_url	The URL for the featured listing being displayed. Use this for creating the link to your featured listing, for example: <code>&lt;a href="{featured_url}"&gt;{listing_title}&lt;/a&gt;</code>
listing_title	Displays the listing's title.
featured_thumb_src	Displays the listing's thumbnail image url for use inside an img tag.
featured_thumb_height	Returns the thumbnail height. For use inside an img tag for the height setting.
featured_thumb_width	Returns the thumbnail width. For use inside an img tag for the width setting.
listing_agent_last_name	Displays the listing agent's last name.
listing_agent_first_name	Displays the listing agent's first name.
listing_agent_thumbnail_#	Displays the listing agent's thumbnail image that is specified in the # place. Whatever number is specified will display the thumbnail that falls in that order
listing_pclass	Displays the name of the property class the listing is assigned to. In cases where the listing is part of multiple classes it will show a the names separated by a comma.
listing_field_FIELDNAME	by using the name of a listing field defined in the listing template editor, you can call the individual field and place it on the page. For example to call the "address" field you would insert {listing_field_address} into your template, which would result in both the field caption and value being placed in your template, eg. "Address: 1600 Penn Ave."
listing_field_FIELDNAME_value	Works the same listing_field_FIELDNAME, except you only get the field value. eg. "1600 Penn Ave."
listing_field_FIELDNAME_caption	Works the same listing_field_FIELDNAME, except you only get the field caption. eg. "Address"
listing_field_FIELDNAME_rawvalue	Works the same listing_field_FIELDNAME, except you only get the raw field value. For example when calling the price field instead of getting "\$150,000" you would simply get "150000".
listing_agent_field_FIELDNAME	by using the name of an agent field defined in the user template editor, you can call the individual field and place it on the page. For example to call the "phone" field you would insert {listing_agent_field_phone} into your template, which would result in both the field caption and value being placed in your template, eg. "Phone: 555-1234"

listing_agent_field_FIELDNAME_value	Works the same listing_agent_field_FIELDNAME, except you only get the field value. eg. "555-1234"
listing_agent_field_FIELDNAME_rawvalue	Works the same listing_agent_field_FIELDNAME, except you only get the raw, unformatted field value. For example when calling the price field instead of getting "\$150,000" you would simply get "150000".
listing_agent_field_FIELDNAME_caption	Works the same listing_agent_field_FIELDNAME, except you only get the field caption. eg. "Phone"
link_to_listing	Places a link to the listing
fulllink_to_listing	absolute link to the listing instead of a relative link. Used for RSS Newsfeeds
money_sign	Adds the configured money sign to the template.
isfavorite	Returns "yes" if listing is marked as a favorite listing by a logged in user and "no" if not marked as favorite or the user is not logged in. Can be used for CSS etc.
featured_large_src	Displays the listing's large image url for use inside an img tag.
featured_large_height	Returns the image height. For use inside an img tag for the height setting.
featured_large_width	Returns the image width. For use inside an img tag for the height setting.
<b>Blocks</b>	
Template tag Blocks are template tags that will have a starting and ending tag. Starting tags will be listed below. Blocks must be closed with <code>{/TEMPLATE_TAG_block}</code> The following template tags are block options:	
featured_listing_block	Block for the each featured listing. The code inside this block will be repeated for each featured listing displayed.
featured_img_block	Block for the image for the featured listing being displayed.
featured_listing_block_row repeat="#"	Block for the featured listing row where rows will be used for displaying featured listings. repeat="#" is the number of times to repeat the content inside the <i>featured_listing_block</i> before starting a new row. When closing this block do not include repeat in the closing tag, i.e. a proper closing would be: <code>{/featured_listing_block_row}</code>
featured_img_large_block	Block for the large image for the featured listing being displayed.

# **SEARCH PROPERTY CLASS TEMPLATES**

(/template/TEMPLATE\_NAME/**FILENAME**)

This template is used to control the layout of the Search Page Property Class selection. The selection of the template is not configurable and this is the template file that is always used for the search page property class selection.

1. **search\_class\_default.html** - This is the search property class template.

## **Search Property Class Template Tags**

Each tag in the template system looks like this **{tag\_name}**.

This template is not a very complicated template and only uses a few template tags to keep it simple. You will need to include your tags inside of a form and field set as well as include the submit buttons to search listings. See the default template for proper coding examples of these fields.

The Following tags are available from the search class templates.

<b>Tag</b>	<b>Description</b>
<i>lang_variable</i>	Displays the value for any defined lang variable.
<i>property_class_checkboxes</i>	Displays the property class checkboxes on the template

# **VIEW USERS TEMPLATE**

*(/template/TEMPLATE\_NAME/FILENAME)*

This template is used to control the layout of the View Users page which displays a list of the agents. This template is not configurable and must be named as below:

1. **view\_users\_default.html** - This is the view users template.

## View Users Template Tags

Each tag in the template system looks like this **{tag\_name}**.

The Following tags are available from the view users template.

<b>Tag</b>	<b>Description</b>
agent_image_thumb_#	Displays the agent's thumbnail image that is specified by the #. <i>note:</i> first image is <b>0</b> , NOT 1
raw_agent_image_thumb_#	Displays the agent's thumbnail image NAME ONLY that is specified by the #. <i>note:</i> first image is <b>0</b> , NOT 1
agent_id	Displays the agent's ID for inclusion in a link. For example: <a href="index.php?action=view_user&user={agent_id}">Click Here</a>
agent_first_name	Displays the agent's first name.
agent_last_name	Displays the agent's last name
agent_contact_link	Renders a link to the agent contact form.
agent_fields	Displays the agent's fields such as phone, address etc.
<b>Blocks</b>	
All code contained on the template inside a block will be controlled according to the block it is contained inside. This is for template designers to control the display of many fields or code by enclosing it in one simple template tag block. A STARTING block will look like a normal template tag but that block must be closed with <code>{/TEMPLATE_TAG}</code> The following template tags are blocks:	
user_block	The block for the display of each user. Code inside the block will be repeated for each user.
agent_image_thumb_#_block	Block for the agent's thumbnail image specified by the #. If there isn't a thumbnail matching that number then code inside this block will not be displayed.

# **VIEW AGENT TEMPLATE**

*(/template/TEMPLATE\_NAME/FILENAME)*

This template is used to control the layout of the Agent Details page. This template is not configurable and must be named as below:

1. **view\_user\_default.html** - This is the view agent template.



## View Agent Template Tags

Each tag in the template system looks like this **{tag\_name}**.

The Following tags are available from the view agent template.

Tag	Description
user_first_name	Displays the user's first name.
user_last_name	Displays the user's last name.
user_images_thumbnails	Displays the thumbnails of the user's images
agent_image_thumb_#	Displays the agent's thumbnail image that is specified by the #. <i>note: first image is 0, NOT 1</i>
raw_agent_image_thumb_#	Displays the agent's thumbnail image NAME ONLY that is specified by the #. <i>note: first image is 0, NOT 1</i>
user_display_info	Displays the user info. All of the fields other than the first name, last name and username.
user_contact_link	Renders a link to contact the agent.
user_listings_list	Displays the agent's listings in a list.
user_hit_count	Displays the hit count for the user.
user_vcard_link	Adds a link to download the agent's information as a vcard.
user_id	Displays the user's id.
user_listings_link	Displays a link to the search results page showing listings from this agent.
user_field_FIELDNAME	by using the name of a user field defined in the agent template editor, you can call the individual field and place it on the page. For example to call the "address" field you would insert {user_field_address} into your template, which would result in both the field caption and value being placed in your template, eg. "Address: 1600 Penn Ave."
user_field_FIELDNAME_value	Works the same as user_field_FIELDNAME, except you only get the field value. eg. "1600 Penn Ave."
user_field_FIELDNAME_caption	Works the same as user_field_FIELDNAME, except you only get the field caption. eg. "Address"
user_field_FIELDNAME_rawvalue	Works the same as user_field_FIELDNAME, except you only get the raw field value. For example when calling a number field instead of getting "150,000" you would simply get "150000".

# **VIRTUAL TOUR (VTOUR) TEMPLATES**

*(/template/TEMPLATE\_NAME/FILENAME)*

This template is used to control the layout of the popup virtual tour page. This template is rather complex due to large javascript requirements in the <head> and other various details.

Which Template is used is determined by the administrator from the configuration screen. Any template named vtour\_templatename.html will be shown as an option in the site configuration.

1. **vtour\_default.html** - This is the default VTour template.
2. **vtour\_classic.html** - This is the "Classic" VTour template. The layout was designed after the original VTour code used in Open-Realty® prior to templating being available.

## VTour Template Tags

Each tag in the template system looks like this **{tag\_name}**.

This template is a little more complex and involved than other Open-Realty® templates due to the large amounts of Javascript that are required to make the Virtual Tours, buttons, controls and other features work properly together.

### Important VTour Template Details

- Inside the <body> tag of your template you must include the tag {onload}. This is required for the PTViewer type of VTours to properly initialize.
- Inside the <head> tag of your template you must include the following tags:
  - {load\_js} **(required to properly load the Javascript into the page)**
  - {license\_tag} **(required per the Open-Realty® license)**
- There are several configuration options available to control the size of the VTour, the size of the popup window and the initial zoom of the VTours. See VTour Config Settings for details.

**In addition to the VTour Template specific tags below, you can also utilize any of the Listing Detail Template Tags. Several of these tags were utilized in the Default VTour Template.**

Tag	Description
vtour	Displays the main VTour Applet on the page. The width and height of the VTour is adjustable in the VTour Config Settings.
vtour_left_button	Displays the Left Nav button for the VTour. You can replace the button with one of your own design as you wish. There are two images that will need to be replaced named vtour_back.gif and vtour_backon.gif. These images are located at /YOURTEMPLATE/images/
vtour_pause_button	Displays the Pause Nav button for the VTour. You can replace the button with one of your own design as you wish. There are two images that will need to be replaced named vtour_pause.gif and vtour_pauseon.gif. These images are located at /YOURTEMPLATE/images/
vtour_right_button	Displays the Right Nav button for the VTour. You can replace the button with one of your own design as you wish. There are two images that will need to be replaced named vtour_forward.gif and vtour_forwardon.gif. These images are located at /YOURTEMPLATE/images/
vtour_zoomin_button	Displays the Zoom In Nav button for the VTour. You can replace the button with one of your own design as you wish. There are two images that will need to be replaced named vtour_zoomin.gif and vtour_zoominon.gif. These images are located at /YOURTEMPLATE/images/
vtour_zoomout_button	Displays the Zoom Out Nav button for the VTour. You can replace the button with one of your own design as you wish. There are two images that will need to be replaced named vtour_zoomout.gif and vtour_zoomouton.gif. These images are located at /YOURTEMPLATE/images/
vtour_select	Displays the VTour drop down select box to select the next Virtual Tour image.
vtour_description	Displays a box for the VTour description.

onload	Adds the proper onload= command to the body tag for PTVIEWER type of VTours. If the VTour is an EGG Solution this tag will be ignored.
money_sign	Adds the configured money sign to the template.
<b>Blocks</b>	
<p>All code contained on the template inside a block will be controlled according to the block it is contained inside. This is for template designers to control the display of many fields or code by enclosing it in one simple template tag block. A STARTING block will look like a normal template tag but that block must be closed with <code>{/TEMPLATE_TAG}</code> The following template tags are blocks:</p>	
vtour_header_block	Place this block around your opening <code>&lt;head&gt;</code> , <code>&lt;html&gt;</code> and <code>&lt;body&gt;</code> tags in your VTour Template. The contents of these tags will be removed if the VTour is being displayed directly on the listing detail page. You must close this block with <code>/vtour_header_block</code>
vtour_footer_block	Place this block around your closing <code>&lt;html&gt;</code> and <code>&lt;body&gt;</code> tags in your VTour Template. The contents of these tags will be removed if the VTour is being displayed directly on the listing detail page. You must close this block with <code>/vtour_footer_block</code>
vtour_content_block	Place this block around any content in your VTour Template that you want removed if the VTour is being displayed directly on the listing detail page. The contents of these tags will be removed if the VTour is being displayed directly on the listing detail page. You must close this block with <code>/vtour_content_block</code>

# **FILE DOWNLOAD TEMPLATES**

(/template/TEMPLATE\_NAME/**FILENAME**)

This template is used to control the layout of the file download section when using the vertical or horizontal templated file list.

Which Template is used is determined by the templated file download template tag used on your site. The templated file download template tags are:

## **For Listing Template:**

**{files\_listing\_NAME}** where *NAME* is the name of the featured listings template included in your template directory. For example using the included templates:

**{files\_listing\_vertical}** will use files\_listing\_vertical.html

**{files\_listing\_horizontal}** will use files\_listing\_horizontal.html

1. **files\_listing\_vertical.html** - This is the vertical listing files template.
2. **files\_listing\_horizontal.html** - This is the horizontal listing files template.

## **For Agent Template:**

**{files\_user\_NAME}** where *NAME* is the name of the featured users template included in your template directory. For example using the included templates:

**{files\_user\_vertical}** will use files\_user\_vertical.html

**{files\_user\_horizontal}** will use files\_user\_horizontal.html

1. **files\_user\_vertical.html** - This is the vertical listing files template.
2. **files\_user\_horizontal.html** - This is the horizontal listing files template.

## File Download Template Tags

Each tag in the template system looks like this **{tag\_name}**.

The Following tags are available from the file download templates.

Tag	Description
lang_variable	Displays the value for any defined lang variable.
file_download_url	The download URL for the file being displayed. This URL is to a download function in Open-Realty®, this will help protect abuse of the file system such as people directly linking to files externally. Use this for creating the link to your files, for example: <code>&lt;a href="{file_download_url}"&gt;{file_filename}&lt;/a&gt;</code>
file_url	The direct download URL for the file being displayed. This URL is a direct link to the file on your server. If you use this download URL then you must remove the included .htaccess files in the files/listings and files/users folders. Be aware that this will allow users to link directly to your files from anywhere and may open your site for abuse by people uploading files to share with others at the expense of your bandwidth. Use this for creating the link to your files, for example: <code>&lt;a href="{file_url}"&gt;{file_filename}&lt;/a&gt;</code>
file_filename	Displays the filename of the file.
file_caption	Displays the caption of the file.
file_description	Displays the description of the file.
file_filesize	Displays the filesize of the file.
file_icon	Displays the url for the file icon for use in a img tag. For example: <code>&lt;img src="{file_icon}" height="{file_icon_height}" width="{file_icon_width}" alt="{file_caption}" /&gt;</code>
file_icon_height	Displays the configured height for the file icons for use in a img tag. For example: <code>&lt;img src="{file_icon}" height="{file_icon_height}" width="{file_icon_width}" alt="{file_caption}" /&gt;</code>
file_icon_width	Displays the configured width for the file icons for use in a img tag. For example: <code>&lt;img src="{file_icon}" height="{file_icon_height}" width="{file_icon_width}" alt="{file_caption}" /&gt;</code>
<b>Blocks</b>	
Template tag Blocks are template tags that will have a starting and ending tag. Starting tags will be listed below. Blocks must be closed with <code>{/TEMPLATE_TAG_block}</code> The following template tags are block options:	
file_block	Block for the each file. The code inside this block will be repeated for each file being displayed. This block is required in order to display the files.
file_block_row repeat="#"	Block for the file row where rows will be used for displaying files in a horizontal layout. repeat="#" is the number of times to repeat the content inside the <i>file_block</i> before starting a new row. When closing this block do not include repeat in the closing tag, i.e. a proper closing would be: <code>{/file_block_row}</code>

# **RSS TEMPLATE**

*(/template/TEMPLATE\_NAME/FILENAME)*

This template is used to control the layout of the XML Data feed for the RSS Newsfeeds. This template is rather complex due to having to follow certain XML structure for proper XML output that can be read by RSS Newreaders. There are also several configuration options on the RSS tab of your site configuration that will effect the output of these newsfeeds. You should be familiar with the RSS Specification if you are attempting to modify the RSS Template. Use the included template as a guide for proper usage of the XML RSS structure.

In order to setup links to your RSS Newsfeeds you will need to add a link to one of the following actions:

**rss\_featured\_listings** - This will provide the user with an RSS Newsfeed of the featured listings. A link to this Newsfeed would look something like this:

```
<a href="index.php?action=rss_featured_listings" >{lang_rss_featured_link_title}</a>
```

**rss\_lastmodified\_listings** - This will provide the user with an RSS Newsfeed of the latest modified listings. A link to this Newsfeed would look something like this:

```
<a href="index.php?action=rss_lastmodified_listings" >{lang_rss_lastmodified_link_title}</a>
```

This template is not a configuration selectable template, your RSS Newsfeeds will always use this specific template file in your selected template directory:

1. **rss.html** - This is the RSS template.

## RSS Template Tags

Each tag in the template system looks like this `{tag_name}`.

**In addition to the RSS Template specific tags below, you can also utilize any of the Listing Detail Template Tags. Several of these tags were utilized in the included RSS Template.**

The Following tags are available from the RSS template:

Tag	Description
rss_title	Places the RSS Title in the RSS Feed. The RSS title is set in the Site Configuration on the RSS tab for the type of RSS feed you are using.
rss_webroot	Places the URL to the RSS Feed you are providing.
rss_description	Places the RSS Description in the RSS Feed. The RSS Description is set in the Site Configuration on the RSS tab for the RSS feed you are using.
rss_listing_guid	Places a Unique identifier for the listing into the RSS feed.
rss_listing_description	Places the listing description into the RSS Feed. The listing description is obtained from the Site Configuration RSS tab in the Listing Description Field for the type of RSS Feed you are providing. HTML and Listing Template Tags can be used in that field.
<b>Blocks</b>	
<p>Template tag Blocks are template tags that will have a starting and ending tag. ALL Code inside a block will be displayed or hidden depending on if the block's conditions are met. Starting tags will be listed below. Blocks must be closed with <code>{/TEMPLATE_TAG_block}</code>. Several of the blocks have a negative switch (!) that will have the opposite condition than its non-negative version. The following template tags are block options:</p>	
rss_listing_block	This block contains the data for listings that will be repeated for every listing in the RSS Feed.



# **NEXT/PREV TEMPLATE**

*(/template/TEMPLATE\_NAME/FILENAME)*

This template is used to control the layout of the Next/Prev section of search results and Listing Details page. The Next/Prev template is not configurable, the site will display one of the two templates depending on the template tag used to call the next/prev function:

1. **next\_prev.html** - This is the next/prev template. Called with **{next\_prev}**
2. **next\_prev\_bottom.html** - This is the next/prev template. Called with **{next\_prev\_bottom}**

## Next/Prev Template Tags

Each tag in the template system looks like this **{tag\_name}**.

The Following tags are available from the Next/Prev template.

Tag	Description
lang_VARIABLE	Displays the corresponding lang variable from the Open-Realty® lang file.
nextprev_num_rows	The number of search results (search result rows)
nextprev_page_type	Displays the lang variable for "next_prev_listing".
nextprev_meet_your_search	Displays the lang variable for "listings_meet_your_search".
nextprev_listing_num_min	Displays the beginning number of the range of listings that are being displayed on the search results.
nextprev_listing_num_max	Displays the ending number of the range of listings that are being displayed on the search results.
nextprev_guidestring_no_action	Displays the search string without the GET action or session id for use in a link to save searches or refine a search. For example: <code>&lt;a href="index.php?action=save_search{nextprev_guidestring_no_action}"&gt;{lang_save_this_search}&lt;/a&gt;</code>
nextprev_guidestring	Displays the full search string with the GET action for use in a link to additional pages of the search results. For example: <code>&lt;a href="index.php?cur_page={nextprev_prevpage}{nextprev_guidestring}" class="bt_pages"&gt; &lt;&lt; &lt;/a&gt;</code>
nextprev_prevpage	Displays the number of the previous page of the search results, can be used in a link to the previous page of the search results. For example: <code>&lt;a href="index.php?cur_page={nextprev_prevpage}{nextprev_guidestring}" class="bt_pages"&gt; &lt;&lt; &lt;/a&gt;</code>
nextprev_disp_count	Displays the number of the page out of the block of 10 pages that we are currently viewing.
nextprev_count	Displays the number of the page out of the block of 10 pages that we are currently viewing. This is for use in a link to the page you want to goto. The link must also include the {nextprev_guidestring}. For Example:

	<code>&lt;a href="index.php?cur_page={nextprev_count}{nextprev_guidestring}" class="bt_pages"&gt;{nextprev_disp_count}&lt;/a&gt;</code>
nextprev_nextpage	Displays the number of the next page of the search results, can be used in a link to the next page of the search results. For example: <code>&lt;a href="index.php?cur_page={nextprev_nextpage}{nextprev_guidestring}" class="bt_pages"&gt; &gt;&gt; &lt;/a&gt;</code>
nextprev_next10page	Displays the number of the first page of the next group of 10 pages, can be used in a link to the next 10 pages (next 100 listings). For example: <code>&lt;a href="index.php?cur_page={nextprev_next10page}{nextprev_guidestring}" &gt;{lang_next_100}&lt;/a&gt;</code>
nextprev_prev10page	Displays the number of the last page of the previous group of 10 pages, can be used in a link to the previous 10 pages (next 100 listings). For example: <code>&lt;a href="index.php?cur_page={nextprev_prev10page}{nextprev_guidestring}" &gt;{lang_previous_100}&lt;/a&gt;</code>
<b>Blocks</b>	
Template tag Blocks are template tags that will have a starting and ending tag. ALL Code inside a block will be displayed or hidden depending on if the block's conditions are met. Starting tags will be listed below. Blocks must be closed with <code>{/TEMPLATE_TAG_block}</code> . Several of the blocks have a negative switch (!) that will have the opposite condition than its non-negative version. The following template tags are block options:	
nextprev_num_of_rows_is_1_block	Block that will be displayed <b>IF</b> there is only one search result row being displayed.
!nextprev_num_of_rows_is_1_block	Block that will be displayed <b>IF</b> there more than one search result row being displayed.
nextprev_show_save_search_block	Block that will be displayed <b>IF</b> this is the search results page to display the save search option. This will not be displayed on other pages that use the next/prev function such as the site log.
nextprev_show_refine_search_block	Block that will be displayed <b>IF</b> this is the search results page to display the refine search option. This will not be displayed on other pages that use the next/prev function such as the site log.
nextprev_is_firstpage_block	Block that will be displayed <b>IF</b> this is the first page of the search results.
!nextprev_is_firstpage_block	Block that will be displayed <b>IF</b> this is NOT the first page of the search results.
nextprev_page_section	Block for the 10 page result set selection. Items in this block will be repeated ten times to create the 10 pages of the search results. There are two blocks that are used inside of

	this block: <code>nextprev_page_current_block</code> and <code>nextprev_page_other_block</code> , see below for details of these blocks
<code>nextprev_page_current_block</code>	Block that is displayed in the <code>{nextprev_page_section}</code> block if, the page number of the search results is the one that we are currently viewing. This is used with the normal template tag, <code>{nextprev_disp_count}</code> to maintain the count and display the count of the page we are on.
<code>nextprev_page_other_block</code>	Block that is displayed in the <code>{nextprev_page_section}</code> block if, the page number of the search results is NOT the one that we are currently viewing. This is used with the normal template tag, <code>{nextprev_disp_count}</code> to maintain the count and display the count of the page we are on.
<code>nextprev_lastpage_block</code>	Block that will be displayed <b>IF</b> this is the last page of the search results.
<code>!nextprev_lastpage_block</code>	Block that will be displayed <b>IF</b> this is NOT the last page of the search results.
<code>nextprev_prev_100_button_block</code>	Block that will be displayed <b>IF</b> we are on a page past the first 10 pages.
<code>!nextprev_prev_100_button_block</code>	Block that will be displayed <b>IF</b> we are NOT on a page past the first 10 pages.
<code>nextprev_next_100_button_block</code>	Block that will be displayed <b>IF</b> there are more than 10 pages left to be displayed.
<code>!nextprev_next_100_button_block</code>	Block that will be displayed <b>IF</b> there are NOT more than 10 pages left to be displayed.
<code>nextprev_clearlog_block</code>	Block that will be displayed <b>IF</b> this is the Admin View Log page and if the user is an admin.

# **LISTINGS NOTIFICATION TEMPLATE**

*(/template/TEMPLATE\_NAME/FILENAME)*

This template is used to control the layout of the listings notification that is sent out by Open-Realty's new listing notification system. The new listing notification emails are to notify users of new listings that match their saved searches.

The Listings Notification system is automated and must be run via a CRON job. See the "Listings Notification" topic in the "Automating Tasks" section of this documentation for instructions on setting up CRON jobs for the Listings Notifications.

Which Notify Listings Template that is used is determined by the site configuration. Any template named `notify_listings_templatename.html` will be shown as an option in the site configuration.

1. **notify\_listings\_default.html** - This is the default listings notification template included with all of the included templates.

## Listings Notification Template Tags

Each tag in the template system looks like this `{tag_name}`.

**In addition to the Listings Notification Template specific tags below, you can also utilize any of the Listing Detail Template Tags. Several of these tags were utilized in the included RSS Template.**

The Following tags are available from the listings notification templates.

Tag	Description
notify_url	Returns the relative URL for a listing.
notify_thumb_src	Returns the URL for the listing thumbnail image
notify_thumb_height	Returns the height of the listing's thumbnail image
notify_thumb_width	Returns the width of the listing's thumbnail image
Blocks	
<p>Template tag Blocks are template tags that will have a starting and ending tag. Starting tags will be listed below. Blocks must be closed with <code>{/TEMPLATE_TAG_block}</code> The following template tags are block options:</p>	
notify_listing_block_row repeat="#"	<p>Block for the file row where rows will be used for displaying files in a horizontal layout. repeat="#" is the number of times to repeat the content inside the <i>file_block</i> before starting a new row. When closing this block do not include repeat in the closing tag, i.e. a proper closing would be: <code>{/file_block_row}</code></p>
notify_listing_block	<p>Block for the each listing. The code inside this block will be repeated for each listing being displayed in the notification. <b>This block is required in order to display the listings.</b></p>
notify_img_block	<p>Block for the listing image. The code inside this block will only be displayed if there is a thumbnail image to be displayed for the listing. If the no image display is turned on then that is considered a thumbnail and this block will be displayed.</p>

# **TAB PANE CODE DOCUMENTATION**

The Tab Pane code used in the Site Configuration is also available for use in any of your Open-Realty® templates. Here is all the documentation on how to use the tabbed template code to create your own content tabs inside your Open-Realty® templates.

## **Tabbed Content**

Tab Panes can be created in any Open-Realty® template file. If creating a Tab Pane in the main template you must ensure it is given a different ID than tab panes used in other template files. If you fail to provide separate ID's the Tab Panes may not appear and function as expected.

### **Create Tab Pane**

The Tab Pane is a division which contains the content of the tab pane known as Tab Pages. Use the following code to create the Tab Pane:

```
<div class="tab-pane" id="tabPanel1">
<script type="text/javascript">tp1 = new WebFXTabPage( document.getElementById(
"tabPanel1" ) );</script>
CONTENT
</div>
```

**div:** The div is the container for the Tab Pane and all of the Tab Pages. The class must be set to "tab-pane". The div id is the ID of this Tab Pane and must be defined. If you have more than one Tab Pane in a template you MUST use different ID's for each Tab Pane.

**Javascript:** The Javascript identifies the div id that will be used for this Tab Pane. In this case tp1 identifies that we are using div id "tabPanel1" for this tab pane. You will use "tp1" in the Tab Page code to add each Tab Page to this Tab Pane.

**CONTENT:** This is where you will place the code for your Tab Pages (see below)

### **Create Tab Pages**

```
<div class="tab-page" id="tabPage1">
<h2 class="tab">Details</h2>
<script type="text/javascript">tp1.addTabPage( document.getElementById( "tabPage1"
) );</script>
CONTENT
</div>
```

**div:** The div is the container for the Tab Page. The class must be set to "tab-page". The div id is the ID of this Tab Page and must be defined. The id for each subsequent Tab Page should be sequential in order to keep things easy to understand.

**h2:** This heading is the title for this Tab Page. The class must be set to "tab".

**Javascript:** This Javascript adds the Tab Page to the Tab Pane. You will need to modify the "tabPage1" with the ID of this Tab Page division.

**CONTENT:** This is where you will place all of your HTML content, Open-Realty® Template tags and other content for this Tab Page.

## **Examples**

For further assistance and examples for creating your own Tabbed content in your Open-Realty® Templates check the coding of the included Tabbed Listing Detail Template.



# AUTOMATING TASKS

## Cron jobs

Cron jobs are a way to trigger a URL or task on a set time basis without having to manually log in to your site. The command that you want your cron job to execute will be different for each task you want to run. The specific command will be noted for each task in this section of the documentation.

Make sure to replace **ORADMINUSER** with your Open-Realty® admin user name, **ORADMINPASSEORD** with your password, and replace **yourdomain.com** with your real domain name and any additional path elements.

***If you are unsure of how to create a cron job with your hosting account, you should contact your host for details as this varies between servers and control panels.***

## Listings Notification

The Listings Notification task should be setup to be run at least once per day. The Listings Notification task will process all users' saved searches and send out any matching new listings added since the last time the task was processed.

It is recommended that you run the task during off-peak hours to reduce load on your site while the task is being run. If you import listings using an import tool such as TransparentRETS® then it is recommended that you run this task after the import of any listings is completed.

### **Listings Notification Cron Command:**

```
curl -d "user_name=ORADMINUSER&user_pass=ORADMINPASSEORD"  
http://www.yourdomain.com/admin/index.php?action=send_notifications
```

### **Listings Notification Manual Trigger:**

If you prefer to process listing notifications manually you can do so with the following URL:

```
http://www.yoursite.com/admin/index.php?action=send_notifications
```

Alternatively you could also add a link to the Listing Notification System in the admin page using the following code:

```
{check_admin}<a href="admin/index.php?action=send_notifications" title="">Process  
Listings Notifications</a>{/check_admin}
```

# DEVELOPER'S GUIDE

## SVN Access and Usage

***SVN Versions are NOT stable releases, and should not be used on live production sites. SVN is useful for testing bug fixes or getting a jump on learning and using new features being added to Open-Realty®.***

***SVN Versions are work in progress and may render your site unusable or cause a loss of data. We encourage anybody that has the knowledge and ability to run a test version of the SVN Build to do so and report bugs.***

These instructions are written for TortoiseSVN because it is what most of us use. If you prefer to use a different SVN client then you are welcome to assuming that you know how to configure it using the information below.

### SVN URL:

<https://svn.transparent-tech.com/open-realty2/trunk/open-realty2>

1) Download a copy of TortoiseSVN and install it. Official download site for the latest version is <http://tortoisesvn.net/downloads>

2) Copy to paste the URL for the latest SVN repository. At present for version 2.x of Open-Realty® this is as follows:

<https://svn.transparent-tech.com/open-realty2/trunk/open-realty2>

Note: In the Support Forum you may see displayed links to this repository or other URLs that have ... as part of the URL. These dots show a shortening of the displayed URL done by the Bulletin Board Software. If you put your mouse cursor on it and look at the URL in the Status bar at the bottom you will probably find that the Full URL is different. It is the full URL you need. To get that, put your mouse cursor on the URL, right click and select 'copy link location'.

3) Open Windows Explorer. Create a new folder on your hard drive (i.e File Menu, - New, - Folder). Select the folder you created and right click it. On the menu that comes up there should be a 'Checkout...' item (probably just above TortoiseSVN). Select this.

4) Click on the 'URL of repository'. Depress CTRL-V (or 'Edit' Menu - 'Paste') - This should paste in the URL that we copied in Step 2 above.

5) Default setup of other items on this page should be fine. Click OK. You will probably have to accept that their certificate isn't correct. This will then download all needed files.

6) A few days later there may be a number of files that have changed. To check which ones have changed, right click on the directory again. This time select 'TortoiseSVN' - 'Check for Updates'. A window will open showing files updated, deleted and added.

7) To update your windows copy, right click on the directory again. This time select 'Update'. (Probably 2 items above TortoiseSVN.)

8) Within Windows Explorer, if you put your mouse on the top of a file you should see a box giving information about the file. The 'Date Modified' in this gives the date and time the file was downloaded to your copy.

9) To view the log, right click on the directory again. This time select 'TortoiseSVN' - 'Show log'. Click on each log header to see the revision information.

10) To learn more about the use of SVN, right click on any file or directory and select 'TortoiseSVN' - 'Help'. The 'Basic Concepts' chapter is an easy read and is needed to understand the difference between this and a traditional source control system.

# Code Standards

- 1) Format your code so that we can read it, please!
- 2) Use tabs for formatting, NOT SPACES. Tabs create smaller files and editors allow developers to view a tab as however many spaces as they prefer. Spaces do not allow this.
- 3) Use ' instead of " for strings. This is a performance issue, and prevents a lot of inconsistent coding styles.
- 4) Comments go on the line ABOVE the code, NOT to the right of the code!
- 5) For each section of code put a section divider with basic explanation of the following code/functions. It should look like this:

```
/******\
*           These functions are used to show a listing           *
\*****/
```

- 6) Do not document every bit of code in comments. PHP is an interpreted language and it will be nasty on performance.
- 7) Use switch statements where many elseif's are going to be used. Switch is faster and we like it better!
- 8) 'If' statements need to use the following format:

```
if ($var == 'example') {
    echo 'This is only an example';
} else {
    echo 'This is not a test. This is the real thing';
}
```

Do NOT make if statements like this:

```
if ($var == 'example'){ echo 'An example'; }
```

All other styles are not to be used. This is it. Use it or we will personally come and nag you to death.

- 9) ALL 'if' statements MUST have matching { } (brackets). Do NOT create 'if' statements like this:

```
if ($a == b)
    dosomething();
```

or:

```
if ($a == b) dosomething();
```

They make the code more difficult to read and follow.

10) class/function format:

```
class testing
{
    function print_to_screen()
    {
        global openrealty, listing_info;

        if ($var == 'example') {
            echo 'This is only an example';
        } else {
            echo 'This is not a test. This is the real thing';
        }
    }
}
```

11) Associative arrays must be written in the following manner:

```
$array = array
(
    'var' => 'value',
    'var2' => 'value2'
);
```

Note that tabs are preferred around the '=>'.

12) Use the long format for <?php. Do NOT use <?.

13) All code should start with 1 tab. Example:

```
<?php
    dosomething();
    if ($a) {
        dosomemorestuff();
    }
```

**NOT:**

```
<?php
dosomething();
if ($a) {
    dosomemorestuff();
}
```

14) Use lower case for variable and function names. No stubbly-case (mixed-case) code.

15) Thanks for following these rules :)

# Virtual Tour Documentation

## Virtual Tours

Open-Realty® supports the following types of Virtual Tours:

- Standard Spherical 360° Panoramic Images in .jpg format (Built in Full Support using PTViewer)
- EGG Solution Virtual Tours using .egg files (Additional file required; See Instructions)

Other, unsupported Virtual Tour formats can be uploaded and used through the creation of an Add-on to handle the display of the unsupported format. You will need to add the file extension and MIME File Type to the Site Configuration in order to upload unsupported formats. Unsupported formats will be displayed as "UNSUPPORTED VTOUR".

See the Add-on Documentation for more information on creating an add-on. Also check the Add-on Releases Forum for any add-ons that may have already been Released to handle your format.

## EGG Solution Virtual Tour Notes:

In order to use the EGG Solution capability of Open-Realty® you will need to copy the EGG Solution viewer applet from your EGG Solution software. This applet is Copyrighted by EGG Solution and we have not been able to gain their permission to include in our distribution of Open-Realty®. We hope we'll be able to get permission in the future.

In your EGG Solution software create a Virtual Tour and Export it to a Web Page. Browse to the location you exported the Virtual Tour to. Copy the following two files to your Open-Realty® root folder:

- Applet.ear
- E3d.jar

## Creating a Virtual Tour

Open-Realty® will not create the virtual tour for you from standard images. You are required to create the virtual tour and upload the resulting images to Open-Realty®.

Standard virtual tours can be created using one of the following methods or possibly other methods not discussed here:

- Digital Camera - Fisheye lens - Tripod and Special Mount - Photo Stitching Software
- Digital Camera - 1 shot panoramic lens\* - Tripod and Standard Mount - Photo Unwrapping software
- Digital Camera - Tripod - Photo Stitching software (note this method will generate an image that can be viewed left to right but not panned up or down and zooming will be limited)

\*There are several different 1 shot lenses on the market. Some of these will not create a panoramic image that is compatible with the Standard Virtual Tours using PTViewer. Such is the case with the EGG Solution System. Many of the 1 shot lenses will come with their own proprietary viewer software and applets. These are not compatible with Open-Realty® however most can easily be made into an Add-On.

## Uploading and Managing Virtual Tours

Once you've created your Virtual tour .jpg images or EGG solution .egg file you will need to upload it for the listing it belongs to.

Login to the control panel using your username and password

Click on **Edit My Listings** or if the user has admin powers **Edit All Listings**

Select the listing you want to edit

On the left side of the window you will see two columns; 1 for Listing Images & 1 for Virtual Tours

Click on Virtual Tours

You will be presented with the virtual tour management screen where you can edit or delete existing virtual tour images and upload new Virtual Tours.

You can either type in the path on your computer to the Virtual Tour or click on **Browse** and find and select your file

Once the file is selected click upload and your file should be uploaded.

You can upload only 1 .egg file for EGG Solution Virtual Tours and you can NOT mix .egg tours with .jpg tours. Standard .jpg images can be uploaded up to the max number as is set in the site configuration.

You will now see your Virtual Tour is uploaded, indicated by a small copy of your jpg image for Standard Virtual Tours and a EGG Solution placeholder image for EGG Solution Virtual Tours. You can edit the Title and Description for the Virtual Tours you've uploaded and they will display under the Tour on Standard Virtual Tours, these fields are not used on EGG Solution Tours although they can be filled out to help you remember details of the tour.

## Distance Search DB

If you want to be able to perform a radius distance search with Open-Realty® then you will need to populate the "zipdist" database table in your Open-Realty® database.

To set up this feature we used zip code and lat/long data provided by:

<http://www.zipcodedownload.com/>

The package you will want to get would be the USA - 5-digit ZIP Code Database, Premium Edition. You can import this database directly into the Open-Realty®

Zip code and Lat/Long data from other providers SHOULD be compatible but you may have to do some manipulation of their data format to import it into Open-Realty®. The zip code data should be imported to the Open-Realty® table: **default\_zipdist**



# ADD-ON INFORMATION

Why should you utilize the add-on system? What can you do with the add-on system? People have made numerous posts on the forums about problems and complaints regarding upgrading from one version of Open-Realty® to a new version. Many of these problems or complaints could have been prevented by the use of the add-on system

The add-on system was well thought out during the development of Open-Realty® and gives great power and flexibility to site developers to add functionality they desire to Open-Realty® without the need to edit the core code of Open-Realty®.

Add-ons don't need to be some large, complex, or spectacular add-on. Add-ons can be as simple as rewriting an existing Open-Realty® function to better suit your taste, design or needs. This will save you from having an unsupported codebase of Open-Realty® that can't easily be upgraded or requires you to reapply modifications you've made to the core code of Open-Realty®.

As an example, the Featured Listings didn't meet the needs of a friend I was designing an Open-Realty® site for recently. Instead of modifying the Featured Listings function in Open-Realty® and having a software that, if upgraded would lose all the changes I've made I rewrote the featured listings function into an Add-on. The new add-on function is now called in his templates instead of the default featured listings functions and he has a software that doesn't require jumping through hoops to upgrade to future bug fixes, or new versions of the software.

Of course, this isn't all that add-ons are for. Some very large and complex add-ons are planned and being worked on as well to add additional functionality to Open-Realty®.

When a change to something is desired, do stop and think about the possibility of using an add-on to do what is required rather than editing the core code of Open-Realty®. This can save you a lot of time in the long run and make your enjoyment of Open-Realty® a lot more fun.

# INSTALLING AN ADD-ON

Open-Realty® Add-ons can be located in the **Add-ons - Releases** sections of the Open-Realty® forums. Add-ons may also be available from other locations on the internet but the forums are the best place to look first. Once you have located a add-on that you want to use in Open-Realty® check to see if documentation for that add-on is provided. This should include instructions for installing the add-on. If instructions for your add-on are provided follow those instructions as there may be special or additional steps required. For a general guide to installing an add-on follow the steps below:

- Download the Add-on - Most add-ons will be distributed in a compressed ZIP file.
- Extract the contents of the compressed file
  - You should be left with a single folder of the Add-on's name that contains the "**addon.inc.php**" file. Look in this folder and verify the `addon.inc.php` file is in this folder. (There may be additional files as well depending on the add-on but **EVERY** add-on must have the **addon.inc.php** file)
    - In some cases your extracted files may be contained in a second sub folder of the same add-on name or may be extracted without being contained inside a folder. Make sure the `addon.inc.php` file is in the proper folder:  
**/addonname/addon.inc.php**
- Upload the add-on folder to your server and place it in the `/addons/` folder of your site. IE: **yoursite.com/addons/addonname**
- Log into the admin area.
- If the add-on has an Admin section, or an Admin icon then you should see that in the add-on section of the Admin index page.
- Go to the Add-on Manager and verify your add-on appears on the list and shows an OK status.

# ADD-ON DEVELOPER GUIDE

## Add-on System Specifications Version 1.1 for Open-Realty®

**Description:** The purpose of this specification is to make a module system that auto-loads any add-ons into Open-Realty®.

Add-ons are very powerful modules for Open-Realty® that will allow most modifications, customizations and additions to be put into an Add-on format instead of having to edit the core code of Open-Realty®. This will make upgrading to future versions of Open-Realty® much easier for the site developers and prevent users from having an unsupported version of Open-Realty® due to code modifications.

All add-ons should have an `addon.inc.php` file. - This file is the core of the add-on system. It is what Open-Realty® will look for to setup and integrate the module. There will be the following defined functions.

**Download the sample Framework Add-on to use as an example of a correctly formatted Add-on.**

## Javascript

As of Open-Realty® 2.1 Add-ons can now load javascript into the template using two new global variables. Add-ons should add any javascript they want to load to the global variables making sure to include the `<script>` `</script>` tags around their javascript. The new variables are:

1: `$jscript` Javascript added to this global variable will be loaded into the main template in place of the `"{load_js}"` template tag.

2: `$jscript_last` Javascript added to this global variable will be loaded into the main template in place of the `"{load_js_last}"` template tag.

You must use `$jscript .= XXX` so the main Open-Realty® javascript is not over written with your code. You MUST define the variable as a global variable in your function(s). You would use these variables in your add-on specific function(s).

## Search Results

If creating an add-on that will be used on the search results page and you would like to get the `listing_ids` for the listings in the search results you can use one of these bits of code:

1: `$matched_listing_ids = search_page::search_results(true);` returns the `listing_ids` for all of the listings in the search results

2: `$matched_listing_ids = search_page::search_results(perpage);` returns the `listing_ids` for the search results with pagination (only the ones displayed on that page of the search results)

# Add-on Functions

All add-on functions should be prefixed with the add-on name. This prefix name should match the add-on folder name, and is represented below by *name\_*.

## name\_install\_addon()

This function should add new database tables and data or modify existing tables and data when necessary. This should be used to store the version number of your add-on in the add-ons table and compare the add-on version to the previously installed version, if any and determine if an installation or an update is needed and then carry out any required database modifications. Here is an example of a proper name\_install\_addon function:

```
function name_install_addon()
{
    $current_version = "1";
    global $conn, $config;
    require_once($config['basepath'].'/include/misc.inc.php');
    $misc = new Misc();
    //Check Current Installed Version
    $sql = 'SELECT addons_version FROM '.$config['table_prefix_no_lang'].'addons
    WHERE addons_name = \''name\';
    $recordSet = $conn->Execute($sql);
    $version = $recordSet->fields[0];
    if ($version == )
    {
        // Preform a new install. Create any needed databases etc, and insert
        version number into addon table.
        $sql = 'INSERT INTO '.$config['table_prefix_no_lang'].'addons
        (addons_version, addons_name) VALUES (\''1\','\'name\')';
        $recordSet = $conn->Execute($sql);
        return TRUE;
    }
    elseif ($version != $current_version)
    {
        //Preform Updates to database based on previous installed version.
        switch($version)
        {
            case '0';
                break;
        } // switch
        return TRUE;
    }
    return FALSE;
}
```

## name\_show\_admin\_icons()

This function should return an array of the html links that should be shown on the administrative page. You can post as simple as a text name of your add-on to show users what's installed, or a link to an administrative page of your add-on. You can display a link using something like this: `index.php?action=name_addonname_admin` You will need to define the action "**name**\_addonname\_admin" in "**name**\_run\_action\_admin\_template" to define it as a `$_GET[]` action for the admin page. Here is an example of a proper `name_show_admin_icons` function:

```
function name_show_admin_icons()
{
    $admin_link = '<a href="index.php?action=addon_name_admin">name</a>';
    return $admin_link;
}
```

## name\_load\_template()

This should return an array with all the template tags for Open-Realty's template engine to parse. If you don't define your template tags for the add-on here they won't be parsed by the template system and properly displayed. The actual replacement of the tags that are defined here is done by the **function name\_run\_template\_user\_fields** Here is an example of a proper `name_load_template` function:

```
function name_load_template()
{
    $template_array = array('addon_name_link');
    return $template_array;
}
```

## name\_run\_action\_user\_template()

This function handles the add-on's `$_GET[]` actions for the **USER** pages. Each get action should have the function to be called defined. The Function must be named using this method: **addon\_name\_description**. Here is an example of proper code for this function:

```
function name_run_action_user_template()
{
    switch ($_GET['action']) {
        case 'addon_name_showpage1':
            $data = name_display_addon_page();
            break;
        default:
            $data = '';
            break;
    } // End switch ($_GET['action'])
    return $data;
}
```

## name\_run\_action\_admin\_template()

This function handles the add-on's \$\_GET[] actions for the **ADMIN** pages. Each get action should have the function to be called defined. The Function must be named using this method: **addon\_name\_description**. Here is an example of proper code for this function:

```
function name_run_action_admin_template()
{
    switch ($_GET['action']) {
        case 'addon_name_admin':
            $data = name_display_admin_page();
            break;
        default:
            $data = '';
            break;
    } // End switch ($_GET['action'])
    return $data;
}
```

## name\_run\_template\_user\_fields()

This function handles all the replacement of {addon\_name\_template\_tags} with the actual content. The tag replacement can call any function that already exists in Open-Realty® or you can create your own **ADD-ON Specific** functions. All tags setup here must also be added to the **name\_load\_template** function in order for Open-Realty® to parse them. Here is an example of proper code for this function:

```
function name_run_template_user_fields($tag = )
{
    switch ($tag) {
        case 'addon_name_link':
            $data = name_display_addon_link();
            break;
        default:
            $data = '';
            break;
    } // End switch ($_GET['action'])
    return $data;
}
```

## name\_addonmanager\_help()

This function provides the information that is displayed in the Add-on Manager help link. There are arrays for Template Tags and Action URLs. There is a variable for a link to the Add-on's documentation URL. Here is an example of proper code for this function:

```
function name_addonmanager_help()
{
    $template_tags=array();
    $action_urls=array();
}
```

```

$doc_url='http://wiki.open-realty.org/framework';
$template_tags['addon_framework_link']='This tag places a link to the
Framework Admin.';
$action_urls['addon_framework_admin']='Displays the Framework Admin page.';
return array($template_tags,$action_urls,$doc_url);
}

```

## **name\_checkupdate\_url()**

This function provides a URL to a plain text file for the add-on manager to check for the latest version of an Add-on. The return string should return a URL to a text file containing the latest version number of the add-on. Here is an example of proper code for this function:

```

function name_checkupdate_url()
{
    $url='http://www.transparent-tech.com/release/framework/version.txt';
    return $url;
}

```

## **name\_update\_url(\$version)**

This function provides a URL to a ZIP file containing the latest version of the add-on. The ZIP file MUST be a ZIP file of ONLY the add-on files and NOT the path of the parent Add-on folder. For Example the Framework add-on should only contain the addon.inc.php file and not the parent /framework/ folder. The return string should return a URL to the latest version of the software. Here is an example of proper code for this function:

```

function name_update_url($version)
{
    $url='http://www.transparent-tech.com/release/framework/latest.zip';
    return $url;
}

```

## **name\_uninstall\_tables()**

This function runs the SQL commands to delete any tables or values that were added during the installation of the add-on. The return string should return TRUE to indicate the add-on's tables were removed properly.

```

function name_uninstall_tables()
{
    global $conn, $config;
    require_once($config['basepath'] . '/include/misc.inc.php');
    $misc = new Misc();

    $sql_uninstall[] = "DELETE FROM " . $config['table_prefix_no_lang'] . "addons
    WHERE addons_name ='framework'";
}

```

```

foreach($sql_uninstall as $elementContents) {
    $recordSet = $conn->Execute($elementContents);
    if ($recordSet === false) {
        echo "<strong><span style=\"red\">ERROR -
            $elementContents</span></strong><br />";
        return FALSE;
    } else {
        return TRUE;
    }
}
}

```

## Add-on Specific Functions

Add-on specific functions are functions that are created specifically for the add-on's use. An example of this would be if a user wanted to create an add-on for a customized Featured Listing layout then the user can write their own Featured listing display function and have it called by the template tag {addon\_superniftyfeaturedlisting\_display} Using all of the above examples of properly formatted add-on functions you would be calling the following Add-on Specific Functions:

```

// Addon Specific Function
function name_display_addon_link()
{
    $display = '<a href="index.php?action=addon_name_showpage1">name Test</a>';
    return $display;
}
// Addon Specific Function
function name_display_addon_page()
{
    $display = 'This is a Addon page';
    return $display;
}
// Addon Specific Function
function name_display_admin_page()
{
    $display = 'This is a Addon page';
    return $display;
}

```

All of these functions have been rolled into a sample add-on called "Framework". You can download the sample Framework Add-on to use as an example or starting point for making your own add-ons.

**Download the sample Framework Add-on to use as an example of a correctly formatted Add-on.**



# LANGUAGE GUIDE

Open-Realty® was designed with multiple language support in mind. Full multi-language support, such as having one site in several languages, is not currently supported. However, we have spent a lot of time in ensuring that Open-Realty® uses language variables for all text on the site. These language variables are defined in "lang" files that can be located at the following path in your Open-Realty® installation:

**include/language/??/versions/original.inc.php**

The ?? in this path is the two letter language code that you have configured in Open-Realty®. For example "en" for English. The file in this path, "**original.inc.php**" contains most of the language variables for Open-Realty®. In addition to the "**original.inc.php**" you will also see additional files named with the Open-Realty® version number they are for and the extension "**.inc.php**". For Example:

**2\_5\_6.inc.php**

**2\_5\_8.inc.php**

Each of these files contains additional language variables that were introduced with those versions of Open-Realty® that are part of the file name. This is to help people keep their sites updated with the latest changes and updates to the language variables.

## Custom changes to language variables

If you find that you want to change the wording of certain Open-Realty language variables then you may use the custom language folder. Custom language files may be placed in the following path:

**include/language/??/custom/**

The ?? in this path is the two letter language code that you have configured in Open-Realty®. For example "en" for English. By default there is a file in this folder named "example.inc.php". This file is an example of how you can use a custom language file.

Create a custom file in this folder and you may then copy language variables from the language files contained in the versions folder. You can make any changes you want to these copied language variables and not have to worry about your custom file being overwritten during upgrades and your customizations being lost.

The language variables in this file will be loaded last and override the default language variables.

## Translate Open-Realty® into a different language

Translating Open-Realty® is not a difficult task, but it is time consuming. Open-Realty® now has numerous regularly maintained languages. If you find that your language is not already in Open-Realty® or is not regularly updated then you can translate the English language files into your language. The files you will need are all of the English files as they are the only language that is guaranteed to be up to date with each release.

Copy all of the English language files into a language folder for your language. The English language files are located in the following path:

**include/language/en/custom/**

The installer language files are maintained in a separate location in the following path:

**install/lang/en/lang.inc.php**

Open the files in your favorite text editor, you will see a list array values for the \$lang array.

For Example:

**\$lang['access\_denied'] = "Access Denied";**

Change the array values only. In the above example, "**Access Denied**" on the variable above, do not modify the array key, which is the **\$lang['access\_denied']** this must remain as it is in the English file.

Once you have translated the Open-Realty® language file please submit it to the Open-Realty® forums so that we can include your translation with the next release. Please post your language file in the **Language File Releases** topic on the Open-Realty® support forums.

# FREQUENTLY ASKED QUESTIONS (FAQ's)

---

## **I'd like to remove the hidden Open-Realty® license from the source, how do I get permission to do this?**

The Open-Realty® license (<http://www.open-realty.org/oslicense.html>) only requires that you purchase a "commercial" license from us, if you want to remove the Attribution Text, which shows up in the Open-Realty® generated html source code.

If that is what you are looking to do, you can order a license using the link below. The cost of the a commercial license to remove the attribution text from Open-Realty®, is \$250.

<http://www.transparent-support.com/billing/cart.php?a=add&pid=28>

If you do not wish to remove the Attribution Text, and only are looking to remove visible references on the template to Open-Realty® or Transparent Technologies you can do so with the open source license.

---

## **I'd like to give something back to the Open-Realty® project, how can I help?**

Thank you for your interest in helping the Open-Realty® project. You can help the Open-Realty® project in many ways such as:

- Make a Donation - Donations can be sent via PayPal to "ryan@transparent-tech.com". Make sure you note that this is a donation for Open-Realty®, or by using the following URL:
    - [https://www.paypal.com/cgi-bin/webscr?cmd=\\_xclick&business=ryan@open-realty.org&item\\_name=Donate%20to%20Open-Realty%20via%20Paypal](https://www.paypal.com/cgi-bin/webscr?cmd=_xclick&business=ryan@open-realty.org&item_name=Donate%20to%20Open-Realty%20via%20Paypal).
  - Donate code for new features or enhancements that you have written for Open-Realty®
  - Provide free Add-ons or Templates for the Open-Realty® community via our forums
  - Provide help and assistance to other users via the support forums.
- 

## **How do I backup my Open-Realty® site?**

All of Open-Realty's listings, users, form data and configuration are stored in the database. So in addition to backing up your Open-Realty® files you will also want to back up your database as well.

1. Backup all of your files. You should have a local copy on your computer from when you uploaded them to your server so it's best to use that as the base for your backup instead of

downloading all your files again. Next step would be to login to your server via FTP and and download the following content to your base Open-Realty® files.

1. `/include/common.php` (Replace the default copy included with Open-Realty® with the one from your server)
  2. The entire contents of your `/images` folder and all subfolders (this contains all your uploaded images and virtual tours)
  3. Any custom templates that you've installed
  4. Any other files that you have modified from Open-Realty® only on the server and not on your local backup copy
2. Backup your database. The easiest way is to use phpMyAdmin or a similar database management software. These are usually installed for most hosting plans and can be accessed through the control panel for your hosting plan (Not the Open-Realty® control panel) To backup using one of these programs you will want to export your entire database that you have setup for Open-Realty®. If you are an advanced user, you can backup just those tables that have the table prefix for your Open-Realty® installation. Save the exported database to a file on your computer with the rest of your Open-Realty® files and you now have a complete backup.
  3. Keep in mind that anytime you add or remove listings your database is being changed. So if you restore an old backup of your database you will have lost any new listings or changes since your last backup.

Many web hosting providers will have an option to download a full site backup from their control panel. These full site backups will include all of the files and the database for your site in a compressed, archived file. This is an excellent way to backup your Open-Realty® site. Contact your host for information and instructions on how to do this on their system if available.

---

## **I lost my password, how do I recover my password?**

Goto the Open-Realty® admin login page. Below the login area is a field to enter your email address and have both your **username** and **password** emailed to you. Enter the email address that you used for your account and click "**Lookup**". Your username and password will be emailed to the email address on your account.

If you do not have access to your email address or don't remember what email address you used for your account, contact the site admin and request that they reset your password or change your email address for you.

---

## **I forgot my admin password, and don't remember or have access to the email address for the admin account. How do I reset the password?**

If emailing the username and password do not work then you can reset the password to the default password if you have access to your host's database administration tools. These instructions assume your host uses cPanel and phpMyAdmin.

Here are the steps to reset the Admin's password for Open-Realty®:

1. Login at "http://www.yourdomain.com/cpanel"
2. Go "mySQL databases"

3. At the bottom of that page click "phpMyAdmin"
4. In the dropdown on the top left side of the page select your Open-Realty® database.
5. On the main section of the screen find the "default\_UserDB" table and click the "Browse" action
6. Find the user you want to reset the password for. Click the "Edit" link next to them
7. In the user\_password field put "5f4dcc3b5aa765d61d8327deb882cf99" (without quotes)
8. Press the go button. Your user password is now reset to "password"

---

## **What is the difference between agents and members?**

Agents can post listings to your site and can also be given more advanced privileges such as editing site pages etc.

Members on the other hand, can not post listings. Members are the people who visit your site, they sign-up to be able to save searches and favorite listings. They can also receive notification by email when new listings that match saved searches are added to the site, you can enable or disable this feature from the site configuration.

---

## **I need to change a word, phrase or how something is said in Open-Realty®, how do I change it?**

Open-Realty® was designed using language variables so that almost all wording on your Open-Realty® site can easily be changed in the language files. The language files contain a large list of all the language variables used on the site and their current values. For details on creating custom language files, editing current language files or changing individual language variables see the Language Documentation section of this documentation.

---

## **I need a feature that Open-Realty® doesn't have, what can I do?**

If you need a special feature that doesn't exist in Open-Realty® you have a few options.

First and foremost you should try searching the Open-Realty® forums to see if anybody has posted an Add-on that includes the feature you want or to see if somebody has posted anything similar.

You can post on our Feature Request system (link in the Support Forums), write an Add-on for Open-Realty® to add the feature to your Open-Realty® site (See Add-on Documentation), post in the Open-Realty® support forum's Work topic to hire somebody to create an add-on, or write the feature into your core code.

Core code modifications are discouraged as they make your Open-Realty® site difficult to upgrade and can create new, unforeseen bugs in your copy of Open-Realty®. If you do choose to code a custom feature into the core code of Open-Realty® that would be beneficial to others you are highly encouraged to post the code to the Feature Request system so that developers can consider it for inclusion in a future release of Open-Realty®.



# TROUBLESHOOTING

---

## **I deleted, lost, overwrote or otherwise messed up my common.php file, what do I do?**

Ideally, you are prepared and have a backup of your common.php file with the proper settings that you can put back on your server. Since you're reading this, that's probably not the case (but remember this for next time!). So, to fix this do the following:

### **Backup your database**

1. Make sure you have the common.php.dist file on your server.
2. Put the install folder back on your server.
3. Run the installer and select the New Installation option
4. Select 'yes' for the Developer Mode install at the bottom.
5. Complete the installation, entering your DB name, username and password.

You will receive a bunch of errors when you complete the installation but it will write a new common.php file for you with your correct Database settings. You can now remove the install folder again and your site should function properly again. Now would be an excellent time to backup your common.php file with all the values stored in it correctly should this happen again.

---

## **I receive 404 Errors when trying to view Open-Realty® pages**

You have enabled Search Engine Friendly (SEF) URLs in your site configuration under the SEO tab. In order for Open-Realty® to function properly with SEF URLs you must also rename the included ".htaccess-or" file to ".htaccess" and upload it to the root of your Open-Realty® installation. See the **Search Engine Friendly URLs and Optimization** section of the "**Installing Open-Realty®**" section of this documentation for more information and details.

---

## **JPG Images are distorted/discolored after i upload them**

If you are using the GD Libs resizing tool (set in your site configuration) make sure your server has version GD Version 2.x or higher. If your server has GD Libs version 2.x or higher then make sure you have configured support for GD Version 2.

1. Log in to the admin panel
  2. Go to the Site Configuration
  3. On the Uploads/Images tab find the setting for **GD Version 2.x** and make sure it is "Yes". Save your changes.
-

## **I see this error on my search results page, Notice: A non well formed numeric value encountered in include/search.inc.php**

This error occurs when you have a listing field setup as a field type of "date" and have a non date value stored in that field. This usually occurs when the field is originally configured as a different field type and then changed to "date". To fix it you need to remove any non date values from the listing field for all of your listings, or change the field type to "text".

---

## **Search results don't display any listings but it says there should be search results**

This problem occurs when you have a default search results sort by field defined that does not contain data or is not available in the property class that you are searching. Make sure that whatever field you have configured as the "**Default Sortby**" in the **Site Configuration** under the **Search Results** tab is available in every property class and always has data in it.

---

## **Search Page is slow to display**

There are a few things you can do to speed up rendering of the search page.

1. In the Open-Realty's Site Config turn "Show Count In Search Options:" off.
2. Increase the "Step" value on all min/max searches so it has to build less options. I recommend setting the "Max Step" value to a low number so that Open-Realty® will not create an unreasonable amount of step values.
3. Look for min/max options where the max value is out of place. Things like 100 bedrooms is almost certainly wrong. Find the listing with the incorrect value and correct it. If you are importing data from your MLS, contact them to have the data correct at the MLS and ask them why they do not have any sort of limits in place to prevent such data entry mistakes from happening.

The search page is rendered dynamically from the Open-Realty® database using values from listings and settings in the site configuration. If, after trying these tips, you still feel your search page is rendering too slow, you may consider creating a hard coded search page. See the **Custom Search Pages** section of the **Search Page Template** documentation section.

---

## **RSS Feed is broken**

RSS Feeds may break in rare circumstances when there are special characters in the Listing Title field and Multi-Byte String support is not enabled in PHP. There should not be any problems without Multi-Byte String support as long as the database collation and Open-Realty® are both set to UTF-8 character sets.



# RECENT CHANGES

## **Open-Realty® Version 2.5.8**

- Rewrite of Open-Realty® documentation
  - Documentation is up to date as of this release.