

Reviewer Plugin

Ultimate Reviews & User Ratings

Version 3.18.0 | Author: [Michele Ivani](#) | Twitter: [@Michele_Ivani](#)

This is an old documentation that is deprecated. Please visit the new online Knowledge Base for all the info about Reviewer plugin.

<http://bit.ly/reviewer-kb>

Index

# Getting started	3
Install the plugin	3
Installation troubleshooting	4
Update the plugin to a new version	4
Register your license	4
Support & FAQs	5
Extended License Notes	6
# The Basics	7
Workflow	7
Plugin customisation	8
How to replace default star icon	8
Enable Google ReCaptcha	8
Enable Google Rich Snippets	9
Reviewer themes	10
Review Box Shortcodes	11
Shortcode Advanced Features	15
Comparison Tables	16
# Widgets	17
Review Boxes widget	17
User Reviews widget	17
# Review Boxes List Shortcode	18
# Translate the plugin	19
# Migrate Plugin's Data	20
Import reviews from an other service	20
# Developers	22
WordPress Custom Hooks	22
Filters	22
Actions	22
Reviewer APIs	22
# Credits	31

Getting started

The document is a how-to guide to install your plugin copy. If you have any questions that are beyond the scope of this help file, please follow the support rules described in the section below. Thanks so much for purchasing this project.

The Reviewer Plugin allows you to insert reviews and comparison tables inside your WordPress blog posts, pages and custom post types in a quick and easy way. If you need to create a gorgeous recap box for your full review post or just a simple but powerful user reviews system, this is the plugin you are looking for.

The plugin flexible structure allows you to create any review type: no limits, just create a template and insert your personal review inside your posts or pages. Users and visitors can leave their reviews on the review box you have created.

The plugin offers different customisable themes so that you can adapt your reviews and comparison tables to your need. Each theme has a responsive layout to adapt to any devices.

Install the plugin

The Reviewer Plugin requires a version of WordPress ≥ 4.2 and a PHP version ≥ 5.3 . So make sure you have the correct environment before installing the plugin.

When you are ready, you can install the plugin in two ways:

- Via WordPress backend: access to the admin area of your WordPress blog and navigate to *Plugins > Add New > Upload*. Click browse and select the **reviewer.zip** from your computer. Hit *Install Now* and the plugin will be uploaded and installed.
- Via FTP: open your FTP client. **Unzip the reviewer.zip** file and upload the non-zipped plugin folder - reviewer - into your WordPress installation folder */wp-content/plugins*.

Once you have installed the plugin, you need to make it active. Go to Plugins and activate the Reviewer Plugin.

Installation troubleshooting

If the installation process returns some errors or you was not able to install the plugin check out the standard errors:

- Your environment - WordPress and PHP versions - does not have minimum requirements.
- You tried to install the zip file you download from CodeCanyon. So unzip it and install the **reviewer.zip** one.

You can contact the Support Team following the rules in the Support section for any issue.

Update the plugin to a new version

If you are going to install a brand new copy of the Reviewer plugin you can skip this section, otherwise **read this notes carefully**.

If you are upgrading from version 1.0 or 1.1, you have to install the version 2.1.1 before installing the latest release. So request the version 2.1.1 via the support section.

If you are upgrading from version 2.0.0 or above you can follow the next steps:

- Backup your blog database using a cPanel tool of your Host like PhpMyAdmin.
- If you translated the plugin, make a backup of your translation files you find at *.../wp-content/plugins/reviewer/languages* folder on your host.
- Deactivate and delete the Reviewer plugin inside the Plugin page of your WordPress blog; **this step does not delete your reviews or plugin data ;-)**
- Install the latest version and activate the plugin again.

Register your license

Version 3.12.0 requires license registration to work; this becomes necessary because the plugin is used illegally by some customers and users.

The registration requires just 30 seconds. Once you have installed and activated the Reviewer plugin, go to *Reviewer > License* page. Fill the form with required information such as your Envato username and purchase code and you will able to enjoy the Reviewer plugin.

If you have any problem during license registration, please contact the Support Team following the rules in the next section. **Please do not post comments on plugin's page at CodeCanyon.**

Support & FAQs

The Reviewer Plugin takes care of its customers, and their satisfaction is one of the primary goals. Before contacting the Support Team, please read the **FAQs** section inside the Support tab of plugin's page at CodeCanyon:

<http://codecanyon.net/item/reviewer-wordpress-plugin/5532349/support>

If you already purchased the plugin you need to register your license - inside the *Reviewer > License* page of your admin area - before contacting the Support Team.

You can contact Support Team via the following methods:

- Send an **email** to support@reviewerplugin.com including your purchase code.
- Send a **message** via the plugin support page at Envato Market.

Please do not post comments on plugin page for support request the Support Team does not manage the comments section.

Each support request is handled within two business days. If you do not receive a reply, please resubmit your inquiry.

Item support includes:

- Availability of the author to answer questions
- Answering technical questions about item's features
- Assistance with reported bugs and issues

Item support does not include:

- Customisation services
- Installation services
- Configuration services
- Compatibility with third-party plugins or themes

Extended License Notes

If you did not purchase an Extended License of Reviewer plugin, you could skip this section.

If you are continuing to read this paragraph, that means you have purchased an Extended License of Reviewer Plugin. Thank you very much! ;-)

A few note when you redistribute/include the Reviewer plugin inside your theme/plugin:

- The Reviewer plugin must be activated before using it since version 3.12.0. The registration process requires your Envato username and the purchase code related to the Reviewer plugin purchase you made. So provide these two pieces of information to your customers and the instructions about Reviewer plugin activation.

The Extended License can be activated in an unlimited number of websites.

- Your customers that use the Reviewer plugin inside your theme or plugin can not have access to the Reviewer Plugin support. They have to contact your support then you can forward the issue to Reviewer plugin support if you are not able to resolve the issue.

Please provide these notes inside your item documentation.

- If you forward the issue to Reviewer Plugin support make sure your Reviewer support license is not expired.

Your collaboration as Theme/Plugin Developer will be appreciated. Thank you.

The Basics

Before describing the workflow and the features of Reviewer Plugin, it's convenient to introduce some terms for a better understanding of next sections.

The main plugin feature is **review boxes**, simple boxes you can insert inside post or pages that contain author review and user reviews. There are three types of review box:

- *Reviewer Box* type is a review box that contains scores, pros and cons, summary and other fields created by the author of the post. Users/visitors can leave their reviews on that box. Think this type as a recap of the full review article the author has written. The *Reviewer Box* must be created and insert in the post/page manually.
- *Users Box* type is a review box specific for site users. The scores - stars, bars, circles - are related to users reviews. If you need a simple but powerful review system, this is the type for you. The *Users Box* must be created and insert in the post/page manually.
- *Auto Box* type has the same behavior of *Users Box* type, but it will be added automatically at the bottom of post/page of particular type. The *Auto Box* can be enabled inside a Reviewer template. If you need a custom position, check the "Advance Shortcode Features" section out.

Workflow

The Reviewer plugin is ready in few simple steps. You can watch the [video preview](#) that shows the basic workflow to create *Reviewer box* otherwise, read the following process.

1. Customize the plugin settings inside *Reviewer > Preferences*.
2. Create a new Template inside *Reviewer > Templates* admin page. The template is a model for your future review boxes. If you decide to use *Auto Box* type you need to enable it inside the template, then skip next step.
3. Create a new post/page or edit an existing one. You will find in the page editor a new section - Reviewer - related to the plugin. Here you can manage the review boxes for the post/page. Choose a review box type and create it. After the configuration, you will find at the bottom of the section a list of WordPress shortcodes that will display the full review box or some of its parts. Just for testing, copy and paste the first one `[rwp_box id="0"]` inside the post/page content and make sure to save/update the post. View the changes.

That's it, with just three steps you will create your first reviews box allowing users and visitors to leave their reviews. You can modify your settings and preferences at any time. The plugin takes care of the changes for you ;-)

Plugin customisation

You can make the look and the layout of review boxes as you need by tweaking settings inside templates or preferences page, but if you need a more customization you can edit the design of reviews box by adding your custom CSS rules inside related text area of *Reviewer > Preferences* page.

If you have not purchased the plugin yet, you can find all the plugin preferences by taking a look at the available [screenshots](#) at Envato Market.

How to replace default star icon

The Reviewer plugin has a nice default icon for ratings, but you can create and upload a custom one. It must follow some required specifications:

- It must be a sprite image exported in PNG format with alpha channel activated.
- Its size must be 48 x 96 pixel with the active and normal state.

You can use the Adobe Photoshop template included with the Reviewer Plugin to create a correct image.

When the Photoshop project is ready, export the image for the Web and upload the new icon inside your Reviewer template.

Enable Google ReCaptcha

Version 3.7.0 replaces default captcha system (secure code for user reviews) with Google ReCaptcha. Now your users can submit reviews more quickly and in a more secure way.

To configure ReCaptcha, you need the **Site Key** and **Secret Key** that Google provides and insert them inside the Reviewer Preferences page. You can get the keys by following the official Google's documentation about ReCaptcha - <https://www.google.com/recaptcha>

- Submit your site to Google Index. <https://www.google.com/webmasters/tools/home> Once you have registered your site go to *Crawl > Fetch as Google* and insert the page you want to index.

You need to wait... Check the site status out after a while. It can take some hours or days.

Why does not my site have the stars inside Google Search?

First, make sure you followed the previous step. It's up to Google to show the stars inside the Google Search. You can not control it. A possible test to check if the stars are available is to make a google search like the following: `site:FULL-URL-OF-THE-PAGE`

Reviewer themes

The plugin includes several themes for your review boxes. You can choose the one you prefer during the Reviewer template creation.

Each theme can be use with any reviews box, but it can have a different behaviour according to the type of box you use.

The themes fields available for the different reviews box types are:

- *Reviewer Box* type: title - criteria score - custom tab - pros & cons - summary - image - custom links - user reviews.
- *Users Box* type: title - criteria score - custom tab - image - custom links - users reviews.
- *Auto Box* type: title - feature image - user reviews.

A hi-resolution preview of all themes is available at plugin page on Envato Market.

Review Box Shortcodes

The Reviewer plugin offers several shortcodes for review boxes. They are available for types [reviewer box](#) and [users box](#) only. Let's describe them one by one.

Release notes:

Reviewer plugin renamed all existing shortcode from version 3.15.0. This was necessary for following the naming convention of WordPress. All previous shortcode names are still valid, so you do not need to change them, but they are deprecated, so please, use the new ones while you are creating new content.

```
[rwp_box id="..."]
```

Display the review box.

Parameters:

id : the review box identifier - default: *none* - **required**

post : the post id of review box - default: *current post id*

Alias (deprecated):

```
[rwp-review id="..."]
```

```
[rwp_reviewer_rating_stars id="..."]
```

Display rating stars about the reviewer score.

Parameters:

id : the review box identifier - default: *none* - **required**

post : the post id of review box - default: *current post id*

size : the size in pixel of stars - default: 24

stars : number of stars for representing the score - default: 5

Alias (deprecated):

```
[rwp-reviewer-rating-stars id="..."]
```

```
[rwp_users_rating_stars id="..."]
```

Display rating stars about the user score.

Parameters:

id : the review box identifier - default: *none* - *required*

post : the post id of review box - default: *current post id*

size : the size in pixel of stars - default: 24

stars : number of stars for representing the score - default: 5

Alias (deprecated):

```
[rwp-users-rating-stars id="..."]
```

```
[rwp_box_recap id="..."]
```

Display some recap sections of review box. The user reviews and form sections are not shown.

Parameters:

id : the reviews box identifier - default: *none* - *required*

post : the post id of review box - default: *current post id*

hide_criteria_scores : set to *true* to hide the criteria scores section - default: *false*

Alias (deprecated):

```
[rwp-review-recap id="..."]
```

```
[rwp_box_criteria id="..."]
```

Display the scores of single criteria.

Parameters:

id : the review box identifier - default: *none* - *required*

post : the post id of reviews box - default: *current post id*

Alias (deprecated):

```
[rwp-review-scores id="..."]
```

```
[rwp_box_reviews id="..."]
```

Display the user reviews only.

Parameters:

id : the review box identifier - default: *none* - **required**

post : the post id of review box - default: *current post id*

Alias (deprecated):

```
[rwp-review-ratings id="..."]
```

```
[rwp_box_form id="..."]
```

Display the form for user rating.

Parameters:

id : the review box identifier - default: *none* - **required**

post : the post id of review box - default: *current post id*

Alias (deprecated):

```
[rwp-review-form id="..."]
```

```
[rwp_user_reviews user="..."] NEW
```

Display all reviews of defined user.

You can find the User ID in the related column of *Users* admin page.

Parameters:

user : the user id identifier - default: *none* - **required**

order : the order of user reviews - default: *latest* or *top_score*

url : custom redirect url for reviews - default: *empty*

limit : how many reviews you want to show - default: *null* or number

stats : if you want to show user statistics - default: *true* or *false*

Full example

```
[rwp_user_reviews user="1" order="top_score", url="http://domain.com/  
reviews-page" limit="10" stats="true"]
```

Let's imagine the scenario in which you want to show the full reviews box (id:0, post:23) in the related post (id:23) and the recap box in another post (id:64). So easy, isn't it?

Insert the `[rwp_box id="0"]` in the post (id:23) then insert the `[rwp_box_recap id="0" post="23"]` in the post (id:64).

[rwp-review]

[rwp-review-recap]

Reviews Box Title



6.3
REVIEWER
A
TAB
5.7
USERS
(123 votes)

PROS

- Lorem ipsum dolor sit amet
- Consectetur adipiscing elit
- Integer eu arcu non erat malesuada

CONS

- Curabitur at varius nulla
- Proin molestie tortor at urna tincidunt

[Custom Link 1](#) [Custom Link 2](#)

SUMMARY

Donec vitae nibh suscipit neque malesuada pulvinar ut et risus. Aenean volutpat et quam vel tincidunt. Sed quis neque gravida, iaculis nisi a, tincidunt massa. Etiam dui tortor, congue in erat quis, porta rutrum lectus. Nullam mattis tortor ornare mauris euismod, id convallis enim porttitor.



[rwp-review-scores]

WHAT PEOPLE SAY... 3 [Leave your review](#)



[rwp-review-ratings]

LEAVE YOUR REVIEW

☆☆☆☆☆

[rwp-review-form]

Shortcode Advanced Features

Paragraph notes: *the following section requires some developers skills so make sure you have access to the source code of WordPress theme and have some knowledge about PHP language.*

The **Auto Box** type allows to automatically add an *Users Box* at the bottom of a post or a page, but if you'd like to render the review box in another place of your post or page using the Auto Box features follow the next steps.

1. Disable Auto Box inside the Reviewer template and copy the Template ID you want to use.
2. Open the file that manages your post/page/custom post type template inside your favourite code editor. Place the cursor in the spot you want to render the reviews box.
3. Paste the following code:

```
<?php echo do_shortcode('[rwp_box id="-1" template="TEMPLATE_ID" ]'); ?>
```

The `id="-1"` is a **wild card** for *Auto Box*. The plugin knows it's *AutoBox* so it generates the id automatically. Replace `TEMPLATE_ID` with the related reviewer template id you want to use.

All process works with the following shortcodes only:

```
[rwp_box]
```

```
[rwp_box_recap]
```

```
[rwp_box_criteria]
```

```
[rwp_box_reviews]
```

```
[rwp_box_form]
```

```
[rwp_users_rating_stars]
```

Comparison Tables

The plugin offers a feature to create tables for comparing reviews boxes. The tables have the same workflow of review boxes and they do not work with *auto box* type.

You can create a new table inside post/page editor using the related box you find in it. Configure the table and paste the shortcode in the post content.

```
[rwp_comparison_table id="..."]
```

Display the comparison table.

Parameters:

id : the table identifier - default: *none* - **required**

post : the post id of table - default: *current post id*

Alias (deprecated):

```
[rwp-table id="..."]
```

Again, if you need to show the same table in a different post, make sure to add the *post* parameter to the shortcode.

The theme previews of comparison tables are shown at plugin page on Envato Market.

Widgets

The plugin offers two WordPress widgets. You can find them in *Appearance > Widgets* page.

Choose the widget you need and drag into a widget area.

Review Boxes widget

The widget displays some information about review boxes - *auto box* type is **not** included - sorted in different ways.

You can filter the review boxes by a template and sort them by:

- Latest boxes
- Top scores boxes by reviewer
- Top rated boxes by users
- Top score boxes by users
- Combo 1 | Average of reviewer and users scores

The widget has 3 gorgeous themes; previews are available at plugin's page.

User Reviews widget

The widget displays a list of users reviews sorted by latest ratings or top scores. You can decide which fields of user review you need to show.

Available fields:

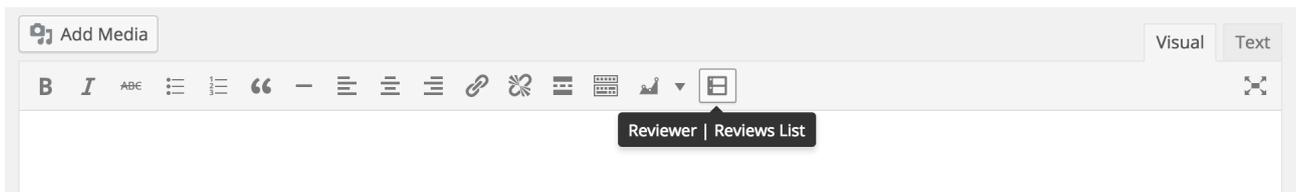
- User Avatar
- User Name
- User Review Title
- User Review Comment
- User Review Score
- Post Title
- Link to the related reviews box

Review Boxes List Shortcode

The shortcode has same features of Review Boxes widget so you can create a custom list of reviews boxes inside post or pages.

The shortcode has a complex structure, so to make the life easier there is a new post editor button that helps you to generate the correct shortcode.

The shortcode can be copy and paste into any post or page.



`[rwp_boxes]`

Display a list of review boxes.

Parameters:

template : reviewer templates ids separated by a colon (:) - default: *all template ids*

title : defines title of the list - default: *none*

sorting : defines sorting mode of the list - default: *latest* - options: *latest, top_score, top_rated, top_users_scores, combo_1*

layout : defines if the list must be displayed inline or full width - default: *auto* - options: *auto, inline*

stars : converts scores number in stars - default: *no* - options: *yes, no*

An example of this shortcode should be:

```
[rwp_boxes template="rwp_template_1234:rwp_template_5678" title="Reviews Boxes List" sorting="latest" stars="yes" layout="auto"]
```

Translate the plugin

The plugin supports WordPress localization. You can translate plugin in your language simply translating a list of labels.

The plugin is localized in English (default) and Italian.

To translate the plugin in a supported languages you need to follow these simple steps:

- Download the free software *Poedit* from its website <http://www.poedit.net>.
- Inside the reviewer folder you can find a subfolder named *languages* that contains all available language files.
- Duplicate the *reviewer-default.po* file and renamed as *reviewer-[language extension].po* You can find your [language extension] on WordPress Codex site at this [link](#).
- Open the new file in *Poedit* and translate all labels you need. You don't need to translate all the labels.
- If you translate the file outside the main *reviewer/languages* folder, you need to upload both *.mo* and *.po* files that Poedit created.
- Once you have completed the process you need to set language of your blog from *Settings > General > Site Language* and you should get your Reviewer Plugin localized.

Migrate Plugin's Data

Since version 2.4 the plugin offers the possibility to import and export the Reviewer data you created. You can easily make I/O operations to backup or to move the Reviewer data to another blog. The version 3.11.0 has the new *Migration* admin page for managing Reviewer backups and migrate user reviews from other review services.

Note: You can import backups made with a plugin version greater or equal to v.3.11.0. So make sure to make a backup if you recently update the plugin.

You can find a copy of all exported files inside the *backup* folder of *reviewer* folder.

Import reviews from an other service

Version 3.11.0 introduces a new tool for migrating user reviews from a review service to Reviewer plugin.

The plugin uses a JSON file with a specific structure that contains all user reviews for the different review boxes you configured. You can find an example of json file in the plugin folder you downloaded from CodeCanyon (*migrate-user-reviews.json*)

First, you need to configure the plugin and create all review boxes you need. The migration process will import the user reviews only.

The migration file contains a list of json objects. Each object must have the following properties:

- *review_post_id*: The numeric ID of the post in which the review box was created .
- *review_box_id*: The numeric ID of the review box. If you are using *Auto Boxes* you can set the value to -1.
- *review_user_id*: The numeric ID of the user who left the review. If the user is not a registered one, you can set the value to 0.
- *review_author*: The name of the review's author. Set the value to an empty string if the user ID is greater then 0.
- *review_author_email*: The email of the review's author. Set the value to an empty string if the user ID is greater 0.

- *review_criteria*: A json object that contains the couple (criterion id, score value). You can find the criterion ids inside the related template of review box. Take a look at example file for all details.
- *review_title*: The title of review.
- *review_comment*: The comment of review
- *review_status*: The review status. Possible values: *publish*, *pending*.
- *review_template*: The template ID related to the review box.
- *review_created_at*: The review date in ISO 8601 format.

All fields are required.

Once your migration file is ready you can validate the json structure at <http://pro.jsonlint.com> and import the file inside the *Reviewer > Migration* admin page of your site.

Note: the process requires some knowledges about json files and dev skills for generating a correct migration file for Reviewer plugin.

Developers

The Reviewer offers some tools to integrate the plugin with your WordPress theme or plugin. This section requires some basic/advanced programming skills.

WordPress Custom Hooks

The plugin has the following WordPress hooks.

Filters

`rwp_before_saving_review`

It will be applied before a new user review is saved on the database. It is useful for make some changes before it will be stored. The callback accept one argument: the review itself.

Actions

`rwp_after_saving_review`

It will be execute before the review process is complete, so after the review is stored in the database successfully.

It accept two arguments: the review itself and the post meta ID that identifies the row in the database.

Reviewer APIs

Since version 2.4 the Reviewer Plugin offers its own API to integrate the plugin functionalities easily with your theme.

Note: the APIs are made for web developers only or for users that has some knowledge of php programming language. They are not shortcodes or standard WordPress features.

The Reviewer API are implemented inside the *RWP_API* php class. The class is already included in your WordPress if the Reviewer plugin is active. All api are *static* methods.

get_reviews_box()

Description

Get review box data of specific post. It returns an empty array if the review box does not exist, otherwise an array with related fields. It works with *Reviewer Box* and *Users Box* types only.

Parameters

The api requires the following parameters:

- `$post_id (int)` The post id that contains the reviews box.
- `$reviews_box_id (int)` The reviews box id. Find it inside the *Reviewer > Reviews Boxes admin page*.
- `$include_users_scores (boolean)` If it is set to *true* the request will include the users scores values and the count. Default: *false*.
- `$include_criteria_labels (boolean)` If it is set to *true* the *review_scores* and *review_users_score* fields will contain the criteria labels. Default: *false*.

Usage

Call the api inside the source code of your theme or plugin.

```
<?php $box = RWP_API::get_reviews_box( $post_id, $reviews_box_id, true ); ?>
```

Output

Note: The *review_users_score* field will be set if the `$include_user_rating` parameter is set to *true*.

```
Array
(
    [review_status] => publish
    [review_type] => PAR+UR
    [review_title] => Reviews Box 0
    [review_title_options] => custom_title
    [review_template] => rwp_template_56b31f19e53cf
    [review_scores] => Array
        (
            [3] => 9
            [1] => 6.5
            [8] => 3
        )

    [review_custom_overall_score] => 5.7
    [review_criteria_source] => reviewer
    [review_custom_tabs] => Array
        (
            [0] => Array
                (
```

```

        [tab_value] => A
        [tab_link] => http://reviewerplugin.com
    )
)

[review_pros] => Lorem ipsum dolor sit amet.
[review_cons] => Lorem ipsum dolor sit amet.
[review_summary] => Lorem ipsum dolor sit amet.
[review_use_featured_image] => no
[review_image] => http://localhost/reviews-box-img.jpg
[review_image_url] => http://reviewerplugin.com
[review_custom_links] => Array
(
    [0] => Array
    (
        [label] => Link
        [url] => #
    )
)

[review_sameas_attr] =>
[review_disable_user_rating] => no
[review_id] => 0
[review_overall_score] => 6.625
[review_users_score] => Array
(
    [scores] => Array
    (
        [3] => 9
        [1] => 3.5
        [8] => 4.5
    )

    [count] => 0
)
)

```

The *review_scores* and *review_users_score* fields will have the following output if the `$include_criteria_labels` parameter is set to *true*.

```

Array
(
    . . .

    [review_scores] => Array
    (
        [3] => Array
        (
            [label] => Criterion B
            [score] => 9
        )

        [1] => Array
        (
            [label] => Criterion A
            [score] => 6.5
        )

        [8] => Array
        (
            [label] => Criterion D
            [score] => 3
        )
    )
)

```

```

        )
    )
    . . .
    [review_users_score] => Array
    (
        [scores] => Array
        (
            [3] => Array
            (
                [label] => Criterion B
                [score] => 9
            )

            [1] => Array
            (
                [label] => Criterion A
                [score] => 3.5
            )

            [8] => Array
            (
                [label] => Criterion D
                [score] => 4.5
            )
        )
        [count] => 0
    )
)

```

Alias

`get_review()` with same parameters.

get_post_reviews_boxes()

Description

Get reviews boxes data of specific post. It returns an empty array if the post has not reviews boxes, otherwise an array with related reviews boxes. It works with *Reviewer Box* and *Users Box* types only.

Parameters

The api requires the following parameters:

- `$post_id (int)` The post id that contains the reviews boxes.
- `$include_users_scores (boolean)` If it is set to *true* the request will include the users scores values and the count. Default: *false*.

Usage

Call the api inside the source code of your theme or plugin.

```
<?php $box = RWP_API::get_post_reviews_boxes( $post_id, true ); ?>
```

Output

Note: The *review_users_score* field will be set if the *\$include_user_rating* parameter is set to *true*. Check the *get_reviews_box* output for the details about reviews box fields.

```
Array
(
    [0] => Array
        (
            [review_id] => 0
            [review_title] => Reviews Box 0
            . . .
        ),
    [1] => Array
        (
            [review_id] => 1
            [review_title] => Reviews Box 1
            . . .
        ),
)
```

Alias

get_post_reviews() with same parameters.

get_reviews_box_users_rating()

Description

Get the users rating of specific reviews box. It returns an array with sigle criterion scores average and the count. It works with *Reviewer Box*, *Users Box* and *Auto Box* types.

Parameters

The api requires the following parameters:

- *\$post_id* (*int*) The post id that contains the reviews box.
- *\$reviews_box_id* (*int*) The reviews box id. Find it inside the *Reviewer > Reviews Boxes* admin page. Use *-1* if you setup *Auto Box* type.

- `$template_id` (*string*) The related template id of reviews box. Required if you setup *Auto Box* type or `$include_criteria_labels` parameter is set to *true*.
- `$include_criteria_labels` (*boolean*) If it is set to *true* the *scores* field will contains the criteria labels. Default: *false*.

Usage

Call the api inside the source code of your theme or plugin.

```
<?php $users_rating = RWP_API::get_reviews_box_users_rating( $post_id,
$reviews_box_id ); ?>
```

Output

```
Array
(
    [scores] => Array
        (
            [3] => 6.5
            [1] => 7
        )
    [count] => 12
)
```

Note: if the `$include_criteria_labels` parameter is set to *true* the output is:

```
Array
(
    [scores] => Array
        (
            [3] => Array
                (
                    [label] => Criterion B
                    [score] => 6.5
                )
            [1] => Array
                (
                    [label] => Criterion A
                    [score] => 7
                )
        )
    [count] => 12
)
```

Alias

`get_review_users_rating()` with same parameters.

get_reviews_box_users_reviews()

Description

Get the users overall rating and single users reviews of specific reviews box. It returns an array with single criterion scores average, the count and all single users reviews. It works with *Reviewer Box*, *Users Box* and *Auto Box* types.

Parameters

The api requires the following parameters:

- `$post_id (int)` The post id that contains the reviews box.
- `$reviews_box_id (int)` The reviews box id. Find it inside the *Reviewer > Reviews Boxes* admin page. Use `-1` if you setup *Auto Box* type.
- `$template_id (string)` The related template id of reviews box.

Usage

Call the api inside the source code of your theme or plugin.

```
<?php $users_reviews = RWP_API::get_reviews_box_users_reviews( $post_id,  
$reviews_box_id, $template_id ); ?>
```

Output

```
Array  
(  
    [overall] => 6  
    [count] => 12  
    [criteria] => Array  
        (  
            [3] => 6.5  
            [1] => 7  
        )  
  
    [reviews] => Array  
        (  
            [0] => Array  
                (  
                    [rating_id] => rwp_rating_56b5c55ebe471  
                    [rating_post_id] => 33  
                    [rating_review_id] => 0  
                    [rating_score] => Array  
                        (  
                            [3] => 4.5  
                            [1] => 8.0  
                        )  
  
                    [rating_user_id] => 0  
                    [rating_user_name] => John  
                    [rating_user_email] => dem@dkc.co  
                    [rating_title] => Lorem ipsum  
                    [rating_comment] => Lorem ipsum  
                )  
        )  
)
```

```

        [rating_date] => 1454756718
        [rating_status] => published
        [rating_template] => rwp_template_56b31f19e53cf
        [rating_socials_url] => Array
            (
                [facebook] => http://. . .
                [twitter] => http://. . .
                [google] => http://. . .
                [email] => mailto:? . . .
                [link] => http://. . .
            )

        [rating_user_avatar] => http://. . .
        [rating_overall] => 6
        [rating_formatted_date] => 23 hours ago
        [rating_helpful] => 0
        [rating_unhelpful] => 0
    )
[1] => Array
    (
        [rating_id] => rwp_rating_56b5c55eer474
        . . .
    )
)
)

```

get_users_reviews()

Description

Get all user reviews of Reviewer plugin filtered by the defined templates ids. It return an array of users reviews. It works with *Reviewer Box* and *Users Box* and *Auto Box* types.

Parameters

The api requires the following parameters:

- `$templates` (*array*) An array will the templates ids.
- `$sort` (*string*) Define the sorting mode. Default: *latest*. Available: *latest*, *top_score*.
- `$limit` (*int*) Define how many users reviews you want to return.

Usage

Call the api inside the source code of your theme or plugin.

```

<?php $reviews = RWP_API::get_reviews_box_users_reviews( array( template_1,
template_3 ), $sort, $limit ); ?>

```

Output

Array

```
(
  [0] => Array
    (
      [rating_id] => rwp_rating_56b5c55e471
      [rating_post_id] => 33
      [rating_review_id] => 0
      [rating_score] => Array
        (
          [3] => 6.5
          [1] => 7.0
        )

      [rating_user_id] => 0
      [rating_user_name] => John
      [rating_user_email] => dem@dkc.co
      [rating_title] => Lorem ipsum
      [rating_comment] => Ciao come va
      [rating_date] => 1454756718
      [rating_status] => published
      [rating_template] => rwp_template_56b31f19e53cf
      [rating_meta_id] => 66
      [rating_post_title] => Test Post
      [rating_user_avatar] => http://. . .
      [rating_overall] => 6
      [rating_formatted_date] => 1 day ago
      [rating_url] => http://domain.com/?rwpurid=rwp_rating_56b5c55e471
      [rating_template_maximum_score] => 10
      [rating_template_minimum_score] => 0
      [rating_template_low_pct] => 30
      [rating_template_high_pct] => 69
      [rating_template_low_score_color] => #56b258
      [rating_template_high_score_color] => #56b258
      [rating_template_medium_score_color] => #56b258
      [rating_template_rate_image] => http://. . .
    )

  [1] => Array
    (
      [rating_id] => rwp_rating_56b5c55e471
      . . .
    )
)
```

Credits

This plugin uses following assets:

- **jQuery Knob Plugin** by *Anthony Terrien* - under MIT License.
- **noUISlider** by Leon Gersen - Free to use in any scenario
- **1942 report** font by Johan Holmdahl - under *Font Squirrel Applications License* - *this free license allows you to embed the fonts in applications and software.* <http://www.fontsquirrel.com/fonts/1942-report>
- **Vue.js** by *Evan You* - under MIT License
- **PhotoSwipe** by Dmitry Semenov - under MIT License
- **ProgressBar.js** by Kimmo Brunfeldt - under MIT License