
pyrs-resource Documentation

Release 0.2.0

Csaba Palankai

August 23, 2015

1	What is this package for	3
2	Nutshell (notice that, it would be the achievement)	5
3	Features	7
4	Installation	9
5	Dependencies	11
6	Important caveats	13
7	The ecosystem	15
8	Contribution	17
9	Contents:	19
9.1	Application	19
9.2	Resource	19
9.3	Response handling	19
9.4	Response handling	19
9.5	Error handling	19
9.6	Hooks	19
9.7	Configuration	19
10	License	21
10.1	Indices and tables	24
	Python Module Index	25

Project homepage: <https://github.com/palankai/pyrs-resource>

Documentation: <http://pyrs-resource.readthedocs.org>

Issue tracking: <https://github.com/palankai/pyrs-resource/issues>

What is this package for

In the python world there are many RESTful framework. Some of them based on Django others are based on Flask. I've tried some but I had the feeling, I want to learn one, the use with Django or Flask or even Odoo. And I don't mention sometimes I found them not flexible enough. So, I've decided write my own independent framework what anybody can use in at least the mentioned 3 different worlds.

Nutshell (notice that, it would be the achievement)

```
from pyrs import resource
from pyrs.resource import GET

class UserResource:

    @GET(response=ArrayOfUserSchema)
    def get_users(self):
        return User.objects.all()

    @PUT(path='/<int:user_id>', response=UserSchema, request=UserSchema)
    def update_user(self, user_id, body):
        user = get_object_or_404(User, pk=user_id)
        user.name = body['name']
        user.email = body['email']
        user.save()
        return user

app = resource.Application()
app.add('/user', UserResource)
```

In this example I've shown Django (like) example. The schema is based on [pyrs.schema](#). Even if I tend to use that framework, you would be able to use any other.

Features

- Using simple classes or even functions (no inheritance)
- Wrapped error handling, errors can be serialised
- Extensible API
- Works with python 2.7, 3.3, 3.4 (tested against these versions)
- Hooks for extending the dispatching process

Installation

```
$ pip install pyrs-resource
```

Dependencies

See *requirements.txt* for details, but mainly depends on [Werkzeug](#). I'm using that project routing capabilities. Also depends on *pyrs.schema* as I mentioned in nutshell section.

Important caveats

This code right now really in beta state. I plan to release soon as possible a completely working code, but right now it's just shaping.

The ecosystem

This work is part of [pyrs framework](#). The complete framework follow the same intention to implement flexible solution.

Contribution

I really welcome any comments! I would be happy if you fork my code or create pull requests. I've already really strong opinions what I want to achieve and how, though any help would be welcomed.

Feel free drop a message to me!

Contents:

9.1 Application

9.2 Resource

9.3 Response handling

9.4 Response handling

```
class pyrs.resource.response.Response (content, app=None, opts=None, request=None)
    Bases: object
    Generic response class
    build()
    setup()
```

9.5 Error handling

9.6 Hooks

9.7 Configuration

This module contains the default configurations. The `pyrs.resource.base.App` config will be based on these values.

```
pyrs.resource.conf.debug = False
    You can get more information in response like traceback and args of exception

pyrs.resource.conf.decorate = '_endpoint'
    This option will be used for decorators. Usage getattr(func, conf.decorate)

pyrs.resource.conf.host = 'localhost'
    Default host for the application
```

`pyrs.resource.conf.inject_app = False`

Enable/disable injecting the `base.App` as keyword argument

`pyrs.resource.conf.inject_app_name = 'app'`

Name used for app injection

`pyrs.resource.conf.inject_auth = False`

Enable/disable injecting the `request.auth` as keyword argument

`pyrs.resource.conf.inject_auth_name = 'auth'`

With this name the auth will be injected

`pyrs.resource.conf.inject_body = True`

Enable/disable injecting the request body

`pyrs.resource.conf.inject_cookies = False`

Enable/disable injecting the cookies

`pyrs.resource.conf.inject_cookies_name = 'cookies'`

With this name the cookies will be injected

`pyrs.resource.conf.inject_path = True`

Enable/disable injecting the path arguments If a name provided the path arguments will be injected as specified

`pyrs.resource.conf.inject_query = True`

Enable/disable injecting the query arguments If a name provided the query arguments will be injected as specified

License

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the
Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

- a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or
- b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

- a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

- a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the Combined Work with a copy of the GNU GPL and this license document.
- c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.
- d) Do one of the following:
 - 0) Convey the Minimal Corresponding Source under the terms of this

License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.
- b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

10.1 Indices and tables

- [genindex](#)
- [modindex](#)
- [search](#)

p

`pyrs.resource.conf`, [19](#)

`pyrs.resource.response`, [19](#)

B

`build()` (`pyrs.resource.response.Response` method), 19

D

`debug` (in module `pyrs.resource.conf`), 19

`decorate` (in module `pyrs.resource.conf`), 19

H

`host` (in module `pyrs.resource.conf`), 19

I

`inject_app` (in module `pyrs.resource.conf`), 19

`inject_app_name` (in module `pyrs.resource.conf`), 20

`inject_auth` (in module `pyrs.resource.conf`), 20

`inject_auth_name` (in module `pyrs.resource.conf`), 20

`inject_body` (in module `pyrs.resource.conf`), 20

`inject_cookies` (in module `pyrs.resource.conf`), 20

`inject_cookies_name` (in module `pyrs.resource.conf`), 20

`inject_path` (in module `pyrs.resource.conf`), 20

`inject_query` (in module `pyrs.resource.conf`), 20

P

`pyrs.resource.conf` (module), 19

`pyrs.resource.response` (module), 19

R

`Response` (class in `pyrs.resource.response`), 19

S

`setup()` (`pyrs.resource.response.Response` method), 19