
django-sysinfo Documentation

Release 1.3.1

Stefano Apostolico

Jul 05, 2018

Contents

1	django-sysinfo	3
1.1	Features	3
1.2	Quickstart	3
1.3	Known issues and limitations	4
2	Installation methods	5
2.1	pip	5
2.2	easy_install	5
2.3	Git	5
3	Configuration	7
3.1	Configure	7
3.2	Sections	7
4	Sample output	11
4.1	JSON	11
4.2	HTML	15
5	Contributing	17
5.1	Types of Contributions	17
5.2	Get Started!	18
5.3	Pull Request Guidelines	19
5.4	Tips	19
6	Credits	21
6.1	Development Lead	21
6.2	Contributors	21
7	History	23
7.1	1.3.1 (23 Feb 2018)	23
7.2	1.3 (10 Feb 2018)	23
7.3	1.2 16 Aug 2017	23
7.4	1.1 14 Jul 2017	23
7.5	1.0 (15 Mar 2017)	24
7.6	0.3 (27 Mar 2016)	24
7.7	0.2 (13 Feb 2016)	24
7.8	0.1.1 (20 Jan 2016)	24

7.9	0.1.0 (15 Jan 2015)	24
-----	-------------------------------	----

Contents:

CHAPTER 1

django-sysinfo



Simple django app to expose system infos like libraries version, database server.

Easy to extend to add custom checks.

1.1 Features

- dump system informations
- check API to add custom checks
- simple echo
- retrieve library version

1.2 Quickstart

Install django-sysinfo:

```
pip install django-sysinfo
```

put it in your *INSTALLED_APPS*:

```
INSTALLED_APPS=[...  
    'django_sysinfo'  
]
```

add relevant entries in your url.conf:

```
urlpatterns = (
    ....
    url(r'', include(django_sysinfo.urls)),
)
```

or customize them:

```
from django_sysinfo.views import http_basic_login, sysinfo



urlpatterns = (
    url('sys/info/$', http_basic_login(sysinfo), name='sys-info'),
    url('sys/version/(?P<name>.*)/$', version, name='sys-version')
)
```

1.3 Known issues and limitations

There are some limitations in the metrics returned by sysinfo, anyway this package is not intended to be used as host/resources monitoring tool.

- Disk space returns device info, any soft limits are ignored
- Memory can be wrong in some virtual environments

1.3.1 Links

Stable			
Development			
Project home page:	https://github.com/saxix/django-sysinfo		
Issue tracker:	https://github.com/saxix/django-sysinfo/issues?sort		
Download:	http://pypi.python.org/pypi/django-sysinfo/		
Documentation:	https://django-sysinfo.readthedocs.org/en/latest/		

Installation methods

Note: It is recommended that you always use the latest release of django-sysinfo with the latest release of Django. If you are using an older version of Django, then please check out the [Compatible Django Versions](#) page for more information.

For information on configuring django-sysinfo, see the getting started guide.

2.1 pip

You can install django-sysinfo into your system, or virtual environment, by running the following command in a terminal:

```
$ pip install django-sysinfo
```

2.2 easy_install

The popular easy_install utility can be used to install the latest django-sysinfo release from the Python Package Index. Simply run the following command in a terminal:

```
$ sudo easy_install django-sysinfo
```

2.3 Git

Using Git to install django-sysinfo provides an easy way of upgrading your installation at a later date. Simply clone the [public git repository](#) and symlink the `src/sysinfo` directory into your `PYTHONPATH`:

```
$ git clone git://github.com/etianen/django-sysinfo.git
$ cd django-sysinfo.git
$ git checkout release-1.9.3
$ ln -s src/sysinfo /your/pythonpath/location/sysinfo
```

3.1 Configure

It possible to include/exclude each session, custom callable and add checks available thru the *check* url

```
SYSINFO = { "os": False,
            "modules": False,
            "python": False,
            "host": False,
            "extra": {},
            "checks": {},
            "project": {
                "mail": False,
                "installed_apps": False,
                "databases": False,
                "MEDIA_ROOT": False,
                "STATIC_ROOT": False,
                "CACHES": False
            }
        }
```

3.2 Sections

3.2.1 Host

```
"host": {
    "hostname": "host1",
    "fqdn": "host1.org",
    "cpus": 4,
    "network": {
        "en0": [
            "192.168.1.100/255.255.255.0"
```

(continues on next page)

(continued from previous page)

```
    ],
    "lo0": [
        "127.0.0.1/255.0.0.0",
        "192.168.66.66/None"
    ]
},
"memory": {
    "percent": "74.5 bytes",
    "used": "5.0 GB",
    "available": "2.0 GB",
    "free": "64.0 MB",
    "total": "8.0 GB"
}
}
```

3.2.2 OS

```
"os": {
    "uname": [
        "Darwin",
        "host1.local",
        "16.4.0",
        "Darwin Kernel Version 16.4.0: Thu Dec 22 22:53:21 PST 2016; root:xnu-3789.41.3~3/
↪RELEASE_ARM_T8020_2",
        "arm64"
    ],
    "name": "posix"
},
```

3.2.3 Python

```
"python": {
    "executable": "/data/ENV/bin/python",
    "version": "2.7.13",
    "platform": "darwin",
    "info": "2.7.13 (default, Mar 10 2017, 12:55:49) \n[GCC 4.2.1 Compatible Apple LLVM
↪8.0.0 (clang-800.0.42.1)]",
    "maxunicode": [
        65535,
        "WARN"
    ]
},
```

3.2.4 Modules

```
"modules": {"alabaster": "0.7.7",
            "apipkg": "1.4",
            "django": "1.9.1",
            "pytest": "2.8.5",
            "xlwt-future": "0.8.0"},
```

3.2.5 Project

```
"project": {
  "current_dir": "/data/PROJECTS/django-sysinfo",
  "tempdir": "/var/folders/vy/jjqmc4bj38z2rj90qzhwsczw0000gn/T",
},
```

installed_apps

```
"installed_apps": [
  [
    "django_sysinfo",
    "0.4a20160403211457"
  ]
]
```

MEDIA_ROOT

```
"MEDIA_ROOT": {
  "path": "/data/PROJECTS/django-sysinfo/tests/demo/demoproject/media",
  "disk": {
    "used": "288.0 GB",
    "free": "176.0 GB",
    "total": "464.0 GB"
  }
},
```

STATIC_ROOT

```
"STATIC_ROOT": {
  "path": "/data/PROJECTS/django-sysinfo/tests/demo/demoproject/static",
  "disk": {
    "used": "288.0 GB",
    "free": "176.0 GB",
    "total": "464.0 GB"
  }
},
```

CACHES

```
"CACHES": {
  "default": {
    "LOCATION": "unique-snowflake",
    "BACKEND": "django.core.cache.backends.dummy.DummyCache"
  }
},
```

Databases

```
"databases": {"default": {"engine": "django.db.backends.postgresql_psycopg2",
                          "host": "127.0.0.1:",
                          "name": "sysinfo",
                          "server": "PostgreSQL 9.4.3 on x86_64-apple-darwin14.3.0,
↳ compiled by Apple LLVM version 6.1.0 (clang-602.0.53) (based on LLVM 3.6.0svn), 64-
↳ bit",
                          "timezone": "UTC",
                          "version": "9.4.3"},
              "sqlite": {"engine": "django.db.backends.sqlite3",
                          "host": ":",
                          "name": ":memory:",
                          "server": "3.8.10.2",
                          "timezone": "UTC",
                          "version": "3.8.10.2"}}
```

Mail

```
"mail" : {"backend": "django.core.mail.backends.locmem.EmailBackend",
          "host": "localhost:25",
          "tls": "false",
          "ssl": "false",
          "status": "OK"}
```

3.2.6 EXTRA

New entries can be added using the extra entry in the configuration.

in your settings

```
SYSINFO = {"extra": { "my-entry", "path.to.myfunc"}}
```

in your path.to package

```
def myfunc(request):
    return "this is my complex function"
```

result

```
"extra" : {"my-entry": "this is my complex function"}
```

CHAPTER 4

Sample output

4.1 JSON

```
{
  "host": {
    "hostname": "host1",
    "fqdn": "host1.org",
    "cpus": 4,
    "network": {
      "en0": [
        "192.168.1.100/255.255.255.0"
      ],
      "lo0": [
        "127.0.0.1/255.0.0.0",
        "192.168.66.66/None"
      ]
    },
    "memory": {
      "percent": "74.5 bytes",
      "used": "5.0 GB",
      "available": "2.0 GB",
      "free": "64.0 MB",
      "total": "8.0 GB"
    }
  },
  "os": {
    "uname": [
      "Darwin",
      "host1.local",
      "16.4.0",
      "Darwin Kernel Version 16.4.0: Thu Dec 22 22:53:21 PST 2016; root:xnu-3789.41.3~
↪ 3/RELEASE_ARM_T8020",
      "x86_64"
    ],
  },
}
```

(continues on next page)

(continued from previous page)

```

    "name": "posix"
  },
  "python": {
    "executable": "/data/ENV/si/bin/python",
    "version": "2.7.13",
    "platform": "darwin",
    "info": "2.7.13 (default, Mar 10 2017, 12:55:49) \n[GCC 4.2.1 Compatible Apple_
↪LLVM 8.0.0 (clang-800.0.42.1)]",
    "maxunicode": [
      65535,
      "WARN"
    ],
  },
  "modules": {
    "alabaster": "0.7.7",
    "apipkg": "1.4",
    "django": "1.9.1",
    "pytest": "2.8.5",
    "xlwt-future": "0.8.0"
  }
},
"project": {
  "current_dir": "/data/PROJECTS/django-sysinfo",
  "tempdir": "/var/folders/vy/jjqmc4bj38z2rj90qzhwsczw0000gn/T",
  "MEDIA_ROOT": {
    "path": "/data/PROJECTS/django-sysinfo/tests/demo/demoproject/media",
    "disk": {
      "used": "288.0 GB",
      "free": "176.0 GB",
      "total": "464.0 GB"
    }
  },
  "STATIC_ROOT": {
    "path": "/data/PROJECTS/django-sysinfo/tests/demo/demoproject/static",
    "disk": {
      "used": "288.0 GB",
      "free": "176.0 GB",
      "total": "464.0 GB"
    }
  },
  "CACHES": {
    "default": {
      "LOCATION": "unique-snowflake",
      "BACKEND": "django.core.cache.backends.dummy.DummyCache"
    }
  },
  "installed_apps": [
    [
      "django_sysinfo", "0.4a20160403211457"
    ]
  ],
  "databases": {
    "default": {
      "engine": "django.db.backends.postgresql_psycopg2",
      "host": "127.0.0.1:",
      "name": "sysinfo",
      "version": "9.6.1",

```

(continues on next page)

(continued from previous page)

```
    "encoding": "UTF8",
    "collate": "en_US.UTF-8",
    "ctype": "en_US.UTF-8",
    "isolation_level": null,
    "timezone": "UTC",
    "info": "PostgreSQL 9.6.1 on x86_64-apple-darwin15.6.0, compiled by Apple LLVM_
↪version 8.0.0 (clang-800.0.42.1), 64-bit"
  },
  "sqlite": {
    "engine": "django.db.backends.sqlite3",
    "host": ":",
    "name": ":memory:",
    "version": "3.14.0"
  }
}
```


4.2 HTML

Mercury

Home » sysinfo

System Information

Project - Python - OS - Modules - Host

Project

current_dir	/data/PROJECT/saxix/mercury	
tempdir	/var/folders/vy/jjgmc4b39a2j09qzlwsczw0060gr/T	
MEDIA_ROOT	/data/PROJECT/saxix/mercury/mercury/media	
STATIC_ROOT	/data/PROJECT/saxix/mercury/src/mercury/staticfiles	
CACHES	default	django.core.cache.backends.locmem.LocMemCache
installed_apps	admin_extra_urls	1.8.0
	django.contrib.admin	2.0.2
	django.contrib.auth	2.0.2
	django.contrib.contenttypes	2.0.2
	django.contrib.humanize	2.0.2
	django.contrib.messages	2.0.2
	django.contrib.sessions	2.0.2
	django.contrib.sites	2.0.2
	django.contrib.staticfiles	2.0.2
	django_sysinfo	1.3.0a0
	jsoneditor	0.0.12
	mercury	0.1
mail	backend	django.core.mail.backends.locmem.EmailBackend
	host	localhost:1025
	TLS	False
	SSL	False
	status	OK

Python

executable	/data/ENV/3sax/bin/python	
version	3.6.3	
platform	darwin	
info	3.6.3 (default, Dec 14 2017, 14:52:42) [GCC 4.2.1 Compatible Apple LLVM 9.0.0 (clang-900.0.39.2)]	
maxunicode	(1114111, 'OK')	

Os

uname	posix.uname_result(sysname='Darwin', nodename='jeeg-3.local', release='17.4.0', version='Darwin Kernel Version 17.4.0: Sun Dec 17 09:19:54 PST 2017; root:xnu-4570.41.2~1/RELEASE_ARM_T8020', machine='arm64t8020')	
name	posix	

Host

hostname	jeeg-3.local	
fqdn	jeeg-3.local	
cpus	4	
network	lo0	127.0.0.1/255.0.0.0
	en0	10.82.54.81/255.255.0.0
memory	total	8.0 GB
	available	2.2 GB
	percent	72.0 bytes
	used	6.3 GB
	free	62.4 MB

Modules

admin-extra-urls	1.8.0
amqp	2.2.2
apipkg	1.4
appnope	0.1.0
argh	0.26.2
arrow	0.12.0
cryptography	2.1.4

4.2. HTML

15

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

5.1 Types of Contributions

5.1.1 Report Bugs

Report bugs at <https://github.com/saxix/django-sysinfo/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

5.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

5.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

5.1.4 Write Documentation

django-sysinfo could always use more documentation, whether as part of the official django-sysinfo docs, in docstrings, or even on the web in blog posts, articles, and such.

5.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/saxix/django-sysinfo/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

5.2 Get Started!

Ready to contribute? Here's how to set up *django-sysinfo* for local development.

1. Fork the *django-sysinfo* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/django-sysinfo.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv django-sysinfo
$ cd django-sysinfo/
$ make develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ make qa
$ py.test tests
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

5.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.7, and 3.5, and for PyPy. Check https://travis-ci.org/saxix/django-sysinfo/pull_requests and make sure that the tests pass for all supported Python versions.

5.4 Tips

To run a subset of tests:

```
$ py.test tests/<test_filename>::<test_name>>
```


CHAPTER 6

Credits

6.1 Development Lead

- Stefano Apostolico <s.apostolico@gmail.com>

6.2 Contributors

None yet. Why not be the first?

7.1 1.3.1 (23 Feb 2018)

- add *extra* section to html output
- minor changes to html output

7.2 1.3 (10 Feb 2018)

- add HTML info page
- Dropped support for old Django (<1.9) versions and Python 2.6
- Django 2.0 support has been added.

7.3 1.2 16 Aug 2017

- fixex python 3.6 compatibility
- new config ‘_ttl’ to set cache max_age attribute

7.4 1.1 14 Jul 2017

- Django 1.11 compatibility
- handle broken database connections
- add new mail server informations
- new ‘checks’ API

- BACKWARD INCONPATIBLE: new config format

7.5 1.0 (15 Mar 2017)

- fixes error in json serialization
- BACKWARD INCOMPATIBLE: by default all the sections are disabled
- allow both string and callable in extra section
- added new configuration parameters
- new management command

7.6 0.3 (27 Mar 2016)

- dropped support Django<1.6
- add `settings.SYSINFO_USERS` to manage access
- default Basic Authentication protected urls
- removed 'sys' prefix from default urlpatterns
- new 'echo' endpoint

7.7 0.2 (13 Feb 2016)

- add some infos
- output sorted to improve readability
- add ability to filter sections (`?s=os,python`)
- add CACHES infos
- removed command line utility

7.8 0.1.1 (20 Jan 2016)

- improved coverage
- fixes typos in copyright

7.9 0.1.0 (15 Jan 2015)

- First release on PyPI.