
Pinry Documentation

Release 1.3.2

Isaac Bythewood

February 25, 2016

| | | |
|----------|-------------------------------|----------|
| 1 | Documentation | 3 |
| 1.1 | Getting Started | 3 |
| 1.2 | Basic Customization | 4 |
| 1.3 | Running On... | 4 |
| 1.4 | Deployment | 5 |
| 2 | Useful Links | 7 |

The open-source core of Pinry, a tiling image board system for people who want to save, tag, and share images, videos and webpages in an easy to skim through format.

1.1 Getting Started

Below are the most basic requirements and a small test to make sure everything is running properly before you get into the heavy lifting. Alternatively you can skip to deployment and use our Dockerfile that will build and run everything for you!

1.1.1 Requirements

Pinry is built on top of Django and optimized to run on a Linux environment. However we have gotten Pinry to work on Windows and Mac as well but it may require some extra digging around configuration. Pinry's Python requirements are all in the `requirements.txt` file and easily installable once you have up a virtual environment. What you need initially:

- Python
- virtualenv
- pip
- Pillow build dependencies or the most recent version installed on your OS and use `virtualenv --system-site-packages` when initiating virtualenv.
- Node
- Bower

After you have all of the above you can skip to Testing and make sure it all works.

1.1.2 Testing

We have many tests built into Pinry to ensure that changes don't break anything. If you are live dangerously and have cutting edge new Pinry features first you can use our master branch for your own instance. We recommend using our tags/versions though.

To run Pinry's tests inside the Pinry repo run:

```
virtualenv .
bin/pip install -r requirements.txt
bin/python manage.py test
```

1.2 Basic Customization

Since we use the standard Django templating system you can edit `pinry/templates` and `pinry/static` to change the overall look and feel of Pinry. It's all basic HTML, CSS and JS built on top of Bootstrap and some custom JavaScript plugins, we don't provide any support for modifications to any of this and future updates of Pinry may need to overwrite your changes so use caution when changing the way Pinry looks.

1.2.1 Custom Settings

We currently have two custom settings you can change in `pinry/settings/__init__.py`:

```
# Set to False to disable people from creating new accounts.
ALLOW_NEW_REGISTRATIONS = False

# Set to False to force users to login before seeing any pins.
PUBLIC = True
```

`ALLOW_NEW_REGISTRATIONS` by default is set to `False` to prevent random people from signing up to your Pinry, to create new private users you can use Django's `createsuperuser`, add them to the database manually or open registrations temporarily while you get your friends/family/coworkers to sign up.

`PUBLIC` by default is set to `True`, if you set to `False` users will have to login to see any of your pins. This is a great way to create a completely private system for a few users or just yourself.

1.3 Running On...

The system that we use and recommend you running Pinry on is Ubuntu. That being said we provide buildout configs and pay very close attention to all other operating systems, you should be able to develop/test/deploy pinry on every platform, we just don't give support for them.

1.3.1 Ubuntu

Ubuntu is pretty simple to get Pinry running get some of our required packages first:

```
sudo apt-get install python-virtualenv git npm
sudo apt-get build-dep python-imaging
sudo npm install -g bower
```

Then you'll need to get Pinry and setup our virtualenv:

```
git clone https://github.com/pinry/pinry.git
cd pinry
virtualenv .
bower install
bin/pip install -r requirements.txt
```

From here you have a full working install of Pinry! You can:

- Run some tests: `bin/python manage.py test`
- Run a development server: `bin/python manage.py runserver`
- Edit the settings files: `pinry/settings`
- Customize the theme: `pinry/templates + pinry/static`

1.3.2 Database Notes

When setting up for the first time you'll need to run syncdb and migrations because we use South:

```
bin/python manage.py syncdb --migrate
```

1.4 Deployment

Deployment for a Django project is easier than most other frameworks and languages but it's harder than a PHP project. We recommend using Docker to deploy Pinry and we already have a Dockerfile created for you to do this. If you'd like to deploy via another method please see [Django's documentation](#) on the subject.

1.4.1 Notes On Deployment

While we don't want to go in depth on Django deployment you will need a few tips for Pinry specific configuration. While most of Pinry acts like a standard Django project we have a special settings setup.

By default Django just has a single `settings.py` file in it's project folder, we deviate from this in that we have a `settings` folder, `pinry/settings`. To change the base settings of Pinry you can play with `pinry/settings/__init__.py` but never import or run directly by pointing to `pinry/settings` or `pinry/settings/__init__.py`, instead use `pinry/settings/development.py` and `pinry/settings/production.py`. For a production deployment you're going to need to edit `production.py` and point that at the correct database and add your own `SECRET_KEY`. Also note that you're going to have to add the setting `ALLOWED_HOSTS` to point at the host names going to your server or Django will block everyone trying to access your site.

1.4.2 Using Docker

Our supported and suggested way to deploy Pinry is using Docker. We provide support and instructions for that over at the [docker-pinry GitHub repository](#).

Useful Links

- Get Pinry: <http://getpinry.com/>
- About Pinry: <http://getpinry.com/about/>
- Pinry Demo: <http://demo.getpinry.com/>
- Pinry Issues: <https://github.com/pinry/pinry/issues>
- pinry/pinry: <https://github.com/pinry/pinry>
- pinry/docker-pinry: <https://github.com/pinry/docker-pinry>
- pinry/getpinry.com: <https://github.com/pinry/getpinry.com>