
SilverFlask Documentation

Release 0.1 alpha

Wolf Vollprecht

May 20, 2015

1	Installation	3
2	The basic building blocks	5
3	User	7

SilverFlask aims to be the first full-featured CMS building on top of solid foundations such as Flask, SQLAlchemy, Jinja2, WTForms and a huge number of plugins built on top of these such as SQLAlchemy-continuum, Flask-User, Flask-Login and countless others.

It wants to be an environment that offers the whole slew of power experienced developers are accustomed to while provide an easy platform for.

SilverFlask is partly named after SilverStripe, one of the best and fully-featured CMS's around in PHP land. However, for my taste it had a few shortcomings. Of course, PHP as a programming language is multitudes less pleasant than python. And unfortunately, SilverStripe is built around a framework that is more or less made only for the CMS. I personally don't like this tight coupling as the framework has not gotten much exposure outside of the SilverStripe CMS world. On the other hand, Flask is a very mature framework, and python is a lovely programming language.

Installation

Warning: SilverFlask is currently in a pre-alpha-stealth state. Don't use it for anything serious. That said, feel free to experiment with it as much as you'd like to!

- Clone repo: `git clone https://github.com/wolfv/SilverFlask`
- **Create a virtualenv with Python 3 (the future is now):**
 - Linux: `export VIRTUALENV_PYTHON=/usr/bin/python3`
 - OSX: `export VIRTUALENV_PYTHON=/usr/local/bin/python3` on OS X with [Homebrew](#)
 - I like virtualenvwrapper, instructions for Ubuntu here: [Link](#)
 - Toggle the virtualenv with `workon <yourvirtualenvname>`
- When in the virtualenv, install all necessary packages via `pip install -r requirements`
- Use the `manage.py` script to create the database (defaults to an sqlite database .db in the app folder: `python manage.py createdb`)
- Start the application server by entering `python manage.py runserver`
- Point your webbrowser to <http://localhost:5000> to visit your first SilverFlask website
- Login as `admin:admin` on <http://localhost:5000/admin>

The basic building blocks

SilverFlask defines a number of basic building blocks for websites.

User

The user model for SilverFlask is already defined and wired up with Flask-Login and Flask-User to provide the whole array of features that you would come to expect from a modern web-application.

Query the current user with `from silverflask.core import current_user`

A Permission model is also included to provide fine-grained access control if needed in `silverflask.models.Role`.