flicket Documentation

Release 0.1.8

Paul Bourne

Contents:

Flicket is a simple web based ticketing system written in Python using the flask web framework which supports English and French locales.

Why Flicket?

I could not find a simple open source ticketing system that I really liked. So, decided to have a crack at creating something written in Python.

1.1 Requirements

1.1.1 Operating System

This will run on either Linux or Windows. Mac is untested.

1.1.2 Python

Python =>3.5 - I have not tested earlier versions of Python 3.

1.1.3 SQL Database Server

Out of the box Flicket is configured to work with MySQL. But there should be no reason other SQLAlchemy supported databases won't work just as well.

Note: When I last tried SQLite I had problems configuring the email settings within the administration settings. You may have to change them manually within SQLite.

1.1.4 Web Server

For a production environment a webserver such as Apache or nginx should be used to serve the application.

1.2 Installation

First read *Requirements*.

It is good practise to create a virtual environment before installing the python package requirements. Virtual environments can be considered a sand boxed python installation for a specific application. They are used since one application may require a different version of a python module than another.

1.2.1 Getting Flicket

The source code for Flicket is hosted at GitHub. You can either get the latest frozen zip file or use the latest master branch.

Zip Package

Download Flicket Dist.zip and unzip.

Master Branch

Get the latest master branch from github using git:

```
git clone https://github.com/evereux/flicket.git
```

Alternatively, download and unzip the master branch zip file.

1.2.2 Installing Python Requirements

Install the requirements using pip::

```
(env) C:\<folder_path>\flicket> pip install requirements.txt
```

1.2.3 Set Up

1. Create your database and a database user that will access the flicket database.

2. If you are using a database server other than MySQL you should change the db_type value within *config.py*. See SQLAlchemy_documentation for options.

3. Create the configuration json file:

   ```
   python scripts/create_json.py
   ```

4. Initialise the database using manage.py from the command line:

   ```
   python manage.py db init
   python manage.py db migrate
   python manage.py db upgrade
   ```

6. Run the set-up script:. This is required to create the Admin user and site url defaults. These can be changed again via the admin panel once you log in:

```
python manage.py run_set_up
```

6. Running development server for testing:

```
python manage.py runserver
```

Log into the server using the username *admin* and the password defined during the setup process.

1.3 Exporting / Importing Flicket Users

1.3.1 Exporting

If you need to export the users from the Flicket database you can run the following command:

python manage.py export_users

This will output a json file formatted thus:

```
[
    {
        'username': 'jblogs',
        'name': 'Joe Blogs',
        'email':'jblogs@email.com',
        'password': 'bcrypt_encoded_string'
    }
]
```

1.3.2 Importing

If you need to import users run the following command:

python manage.py import_users

The file has to formatted as shown in the Exporting example.