ADR Colorado College Utilities Documentation

Release 0.2.0

Jeremy Nelson

February 19, 2016

Contents

All technical documentation is located at https://readthedocs.org/projects/adr-cc-utilities-web-app

CHAPTER 1

WARNING!!!

This web app has read/write access to your repository and is designed to be run internally on your institution's network. **DO NOT** allow open-web access to this web app unless you are prepared for the consequences.

CHAPTER 2

Using Windows Preloaded Environment

To run this web app from pre-built Windows 32-bit environment, download the adr-cc-utilities-win32-env.zip and unzip to a working directory.

Go to the Configuration section below and follow the directions to configure the **fedora.cfg** and **form-variables.cfg** to properly run this web app. To run the web app from the windows environment, open a Windows command-line and follow these directions (assuming *adr-cc-utilities-win32-env.zip* was saved and unzipped from current user's **Downloads** directory)

```
C:\Users\current_user\Downloads>cd adr-cc-utilities-win32-env
C:\Users\current_user\Downloads\adr-cc-utilities-win32-env\start.bat
Running ADR-CC-Utilities Locally in Dev mode
To stop server, use ctrl-c key combination
Connect with your web-browser at http://localhost:8003/
```

To close the running web app, use the *ctrl-c* key combination in the command line window that is running. You should now be able to connect to the web app from your browser at http://localhost:8003/.

CHAPTER 3

Installing from Source

To use this web app on Linux or Macintosh, follow these directions to install using either *git* or by downloading a source zip file.

To install this web app, you can either fork/clone the project's source code from https://github.com/jermnelson/adr-cc-utilities/:

```
$ git clone https://github.com/jermnelson/adr-cc-utilities
```

download a zip file and unzip the file to create the app's working directory.:

```
$ unzip adr-cc-utilities-master.zip adr-cc-utilities
```

Change directories to the new `adr-cc-utilities` directory and run this command to install the Python modules required by this web app

```
$ cd adr-cc-utilities
$ python setup.py install
```

CHAPTER 4

Configuration

To use this web app, you'll need to first copy create two Flask configuration files: **fedora.cfg** and **form-variables.cfg** both located in `adr-cc-utilities` directory.

From Windows

```
$ copy example-fedora.cfg fedora.cfg
$ copy example-form-variables.cfg form-variables.cfg
```

For Linux, MacOS, or Window Powershell

```
$ cp example-fedora.cfg fedora.cfg
$ cp example-form-variables.cfg form-variables.cfg
```

You then need to change the following variables in each of the configuration files to match your Fedora Commons Server root, location, username, and password along with Institutional details for the batch Fedora Objects ingestion.

4.1 fedora.cfg

Variables for **fedora.cfg** are listed in the configuration section in the Flask Fedora Commons Documentation as well.

FEDORA_ROOT	‘http://fedora.host.name:8080/fedora/‘
FEDORA_USER	‘user’
FEDORA_PASSWORD	‘password’
FEDORA_PIDSPACE	‘changeme’
FEDORA_TEST_ROOT	‘http://fedora.host.name:8180/fedora/‘
FEDORA_TEST_PIDSPACE	‘testme’

4.2 form-variables.cfg

Variables in **form-variables.cfg** provide common information, such as INSTITUTION_NAME, for the forms used in this web app These variables (and examples values) are :

SECRET_KEY	‘’
CONTENT_MODELS	‘[”adr:adrBasicObject”, “adr:adrETD”]’
INSTITUTION_NAME	‘Any Memory Institution’
LOCATION	‘City, State’
RIGHTS_STATEMENTS	‘Copyright by Any Memory Institution, all rights reserved’

The *SECRET_KEY* variable is needed by the WTForms, put a random string of characters.

CHAPTER 5

Running as a local web app

To run this web app from the command line in either **development** or **production** mode, use the following command with these options.

Run in development mode with default **host** of *0.0.0.0* and default **port** of 8003

```
$ python server.py dev
```

Run in production mode with optional *–host* and *–port* options

```
$ python server.py prod --host=localhost --port=8080
```

You should now be able to use your web browser of choice to access the web app. In the default **development** or **production** modes, load http://localhost:8003/ into the browser's address bar.

CHAPTER 6

Classes and Methods

CHAPTER 7

Indices and tables

- genindex
- modindex
- search