pyPreservica.Gov

James Carr

Oct 12, 2023

TABLE OF CONTENTS:

Release v1.0.0.

pyPreservica.Gov Python Module For Harvesting public Modern.Gov Committee and Meeting Records into Preservica for Long Term Preservation.

pyPreservica.Gov is a Python Module which allows Preservica users to automatically harvest public records from the Civica Modern.Gov Content Management system directly into a Preservica Long Term Digital Preservation platform

pyPreservica.Gov is an open source 3rd party application and is not affiliated with Preservica Ltd

There is no support for use of the library by Preservica Ltd. For support see Support

TABLE OF CONTENTS:

WHY SHOULD I USE THIS?

pyPreservica.Gov is a Python module which allows you to harvest metadata and content from Civica Modern.Gov meeting management software.

pyPreservica.Gov will harvest Modern.Gov committees and meetings and store the metadata and meeting outputs in Preservica

PYPRESERVICA.GOV FEATURES

- Fetch Committees and Committee metadata
- Fetch Meeting outputs such as Meeting Agenda's, Minutes and Appendices
- Index Committee and Meeting metadata

PIP INSTALLATION

pyPreservica.Gov is available from the Python Package Index (PyPI)

https://pypi.org/project/pyPreservicaGov/

pyPreservica.Gov is built and tested against Python 3.8 and above. Older versions of Python may not work.

To install pyPreservica.Gov, simply run this simple command in your terminal of choice:

```
$ pip install pyPreservicaGov
```

or you can install in a virtual python environment using:

```
$ pipenv install pyPreservicaGov
```

pyPreservica.Gov is under active development and the latest version is installed using

```
$ pip install --upgrade pyPreservicaGov
```

GET THE SOURCE CODE

pyPreservica.Gov is developed on GitHub, where the code is always available.

You can clone the public repository

```
$ git clone git://github.com/carj/pyPreservica.Gov.git
```

CONTRIBUTING

Bug reports and pull requests are welcome on GitHub at https://github.com/carj/pyPreservica.Gov

USAGE

Once installed you can run pyPreservica.Gov directly from the command line

```
$ python -m  pyPreservicaGov
```

The first time the module is run, it will create a credentials.properties file in the local directory and then exit

Edit the credentials.properties, the first section should contain your Preservica username, password and hostname of the Preservica system you would like to connect to.

The second section contains the security tag which should be assigned to the records as they are ingested. For example records which can be published immediately on the Preservica access portal should be set to "public".

The site_name parameter should point to the URL of the Modern.Gov system you would like to harvest.

parent.folder should be the UUID of the Preservica collection the Committee folders will be ingested into.

The committee.FromDate and committee.ToDate parameters specify the date range of meetings which will be harvested.

```
[credentials]
username=test@test.com
password=1234567
server=uk.preservica.com

[Modern.Gov]
security.tag=open
site_name=https://democracy.local_authority.gov.uk/
parent.folder=372d2881-7cce-4c2e-99f7-386c3cf4a922
committee.FromDate=01/01/1980
committee.ToDate=01/01/2024
```

Once the credentials.properties has been updated you can run pyPreservica.Gov again directly from the command line to start the harvest process.

```
$ python -m  pyPreservicaGov
```

INDEX

- genindex

PYTHON MODULE INDEX

p

M

P