matyan Documentation

Release 0.4.2

Artur Barseghyan <artur.barseghyan@gmail.com>

Dec 25, 2019

Contents

Generate changelog from Git commits.

CHAPTER 1

Prerequisites

- Python 3.6, 3.7 and 3.8

CHAPTER 2

Documentation

Documentation is available on Read the Docs.

CHAPTER 3

Installation

Latest stable version on PyPI:

```
pip install matyan
```

CHAPTER 4

Usage

4.1 Basic usage

See Basic concepts section to get impression on possible commit methodology and assumptions taken.

Generate changelog:

```
generate-changelog
```

Generate changelog skipping orphaned commits:

In some cases you would only want to show what has been done with tickets and skip all non-ticket related commits.

```
generate-changelog --no-other
```

Generate changelog between two releases:

In other cases you would want to show what has been done since last release. The following example would generate changelog since version 0.0.1 to version 0.0.3.

```
generate-changelog 0.0.1..0.0.3
```

Generate changelog between two branches:

Sometimes you just need to show the changes made on acceptance since last production release. The following example would generate changelog with changes that are on acceptance branch and not yet in master.

```
generate-changelog master..acceptance
```

Generate changelog with releases info shown

```
generate-changelog --show-releases
```

Generate changelog between releases with releases info shown

```
generate-changelog 0.0.1..0.0.3 --show-releases
```

Generate changelog between branches with releases info shown

```
generate-changelog master..dev --show-releases
```

Generate changelog for the latest release with releases info shown

```
generate-changelog --latest-release --show-releases
```

Generate changelog with headings only (no commit messages) and releases info shown

```
generate-changelog --headings-only --show-releases
```

Generate changelog between two branches, show unreleased changes only:

```
generate-changelog master..acceptance --show-releases --unreleased-only
```

4.2 Rendering

The following renderers are implemented:

- Markdown
- RestructuredText
- Historical Markdown (for compatibility with `matyan` versions prior to 0.4).

4.2.1 Markdown

```
generate-changelog --show-releases --renderer=markdown
```

4.2.2 RestructuredText

```
generate-changelog --show-releases --renderer=rest
```

4.2.3 Historical Markdown

```
generate-changelog --show-releases --renderer=historical-markdown
```

4.3 Jira integration

It's possible to fetch ticket title and description from Jira. In order for it to work, you should provide a `fetch-title` and `fetch-description` arguments.

The following needs to be added to your `.matyan.ini`:

```
[Settings]
fetchDataFrom=Jira
```

In addition to that, you should put valid Jira credentials into your global `.matyan.ini` configuration file.

Command to run:

```
generate-changelog --show-releases --fetch-title --fetch-description
```

Have in mind, that `matyan` shall be installed with `jira` option.

```
pip install matyan[jira]
```

Alternatively, make sure `atlassian-python-api` is installed.

```
pip install atlassian-python-api
```

4.4 Examples

See the output directory for examples.

CHAPTER 5

Configuration

In order to customize names and texts, add a `.matyan.ini` in your project directory, from which you will be running the `generate-changelog` command.

Sample configuration:

```
[BranchTypes]
feature: Feature
bugfix: Bugfix
hotfix: Hotfix
deprecation: Deprecation

[OtherBranchType]
other: Other

[Unreleased]
unreleased: Unreleased

[IgnoreCommits]
exact: more
       clean up
       code comments
       more on docs
       repo
       working
       more on
       wip
       commit
prefix: more on
        continue on
```

Note, that placing `.matyan.ini` into the home root will make that configuration global for all projects. That however could be handy, since local `.matyan.ini` files simply override the global ones. For example, you could use global configuration for storing Jira credentials.

```
[Jira]
url:https://barseghyanartur.atlassian.net/
username:user@domain.com
token:abcd1234
```

CHAPTER 6

Tips and tricks

6.1 Write to file

```
generate-changelog --show-releases 2>&1 | tee changelog.md
```

6.2 Create initial config file

```
matyan-make-config
```

CHAPTER 7

Testing

Simply type:

```
./runtests.py
```

Or use tox:

```
tox
```

Or use tox to check specific env:

```
tox -e py38
```

CHAPTER 8

Debugging

Sometimes checking logs could be handy. `Matyan` logs are stored in the directory, from which you are running the `generate-changelog` (or any other `Matyan`) command.

```
tail -f /path/to/your/matyan.log
```

If you want to modify current logging, use `MATYAN_LOGGING_CONFIG` environment variable.

Default configuration:

```
DEFAULT_LOGGING_CONFIG = {
    'version': 1,
    'disable_existing_loggers': False,
    'root': {
        'level': 'WARNING',
        'handlers': ['file'],
    },
    'formatters': {
        'verbose': {
            'format': '{levelname} {asctime} {module} {process:d} {thread:d} '
                      '{message}',
            'style': '{',
        },
        'simple': {
            'format': '{levelname} {message}',
            'style': '{',
        },
    },
    'handlers': {
        'console': {
            'level': 'WARNING',
            'class': 'logging.StreamHandler',
            'formatter': 'simple'
        },
        'file': {
```

(continues on next page)

(continued from previous page)

```
            'level': 'WARNING',
            'class': 'logging.handlers.RotatingFileHandler',
            'filename': os.path.join(os.getcwd(), "matyan.log"),
            'maxBytes': 1048576,
            'backupCount': 99,
            'formatter': 'verbose',
        },
    },
    'loggers': {
        'matyan': {
            'handlers': ['file'],
            'propagate': True,
        },
    },
}
```

CHAPTER 9

Writing documentation

Keep the following hierarchy.

```
=====
title
=====

header
======

sub-header
----------

sub-sub-header
~~~~~~~~~~~~~~

sub-sub-sub-header
^^^^^^^^^^^^^^^^^^

sub-sub-sub-sub-header
++++++++++++++++++++++

sub-sub-sub-sub-sub-header
**************************
```

CHAPTER 10

License

GPL-2.0-only OR LGPL-2.1-or-later

CHAPTER 11

Support

For any issues contact me at the e-mail given in the *Author* section.

CHAPTER 12

Author

Artur Barseghyan <artur.barseghyan@gmail.com>

CHAPTER 13

Docs

Contents:

13.1 Use cases and basic concepts

If the following applies to you, `matyan` could help:

- Project releases (tags) are numbered according to the semantic versioning or sequence based identifiers.
- Project follows the DTAP.
- Testing, acceptance and production branches (hereafter referred as TAP branches) are protected.
- Direct commits to TAP branches are forbidden.
- All commits to TAP branches are made by pull requests.
- JIRA (or a similar tool) is used for handing project tickets.
- Pull requests are merged using GitHub or BitBucket web interface.

13.1.1 Sample use-case

13.1.1.1 The use-case

- JIRA is used for issues.
- All commits are prefixed with ID of the JIRA issue: for example, *MSFT-1234* or *NVDA-1234* (where first four letters identify the client commit was done for, it's pattern).
- There are 3 main (protected) branches: *dev*, *staging*, *master*. Direct commits to any of the 3 are forbidden. Any feature/bugfix comes via merge request.
- All branches do have meaningful prefixes. Example, *feature/MSFT-1234-Title-of-the-issue* or *bugfix/MSFT-1236-prevent-duplicate-postal-codes*.

- Release flow is *dev -> staging -> master*.

13.1.1.2 Sample commits

Consider the following commits into the dev branch:

branch: bugfix/MSFT-1240-LinkedIn-authentication-failing

- MSFT-1240 Fix package configuration.
- MSFT-1240 Update authentication pipeline.

branch: deprecation/MSFT-1239-Deprecate-Python2

- MSFT-1239 Deprecate Python2.
- MSFT-1238 Add initial MyPY setup.

branch: feature/MSFT-1238-Token-authentication

- MSFT-1238 Implement token authentication.
- MSFT-1238 Update authentication docs.

branch: feature/MSFT-1237-Improve-document-sharing

- MSFT-1237 Improve document sharing. Add option to share via GDrive.

branch: bugfix/MSFT-1236-prevent-duplicate-postal-codes

- MSFT-1236 Normalise postal codes for German addresses.
- MSFT-1236 Normalise postal codes for US addresses.
- MSFT-1236 Make postal code field unique for the country.

branch: deprecation/MSFT-1235-deprecate-old-api

- MSFT-1235 Deprecate API v 2.0.
- MSFT-1235 Update docs.

branch: feature/MSFT-1234-car-type-suggester

- MSFT-1234 Initial car type suggester implementation.
- MSFT-1234 Add insurance amount indication based on car weight.

13.1.1.3 Sample releases

All commits have been finally merged into master.

Releases have been made in the following way:

0.1

- Merged issues MSFT-1234, MSFT-1235 and MSFT-1236

0.2

- Merged issues MSFT-1237 and MSFT-1238

Yet unreleased features/branches

- MSFT-1239 and

13.1.2 Sample changelog output

The generated change log would look as follows:

```
### 0.2

**Features**

*MSFT-1238 Token-authentication*

- Implement token authentication.
- Update authentication docs.

*MSFT-1237 Improve document sharing*

- Improve document sharing. Add option to share via GDrive.

### 0.1

**Bugfixes**

*MSFT-1236 Prevent duplicate postal codes*

- Normalise postal codes for German addresses.
- Normalise postal codes for US addresses.
- Make postal code field unique for the country.

**Deprecations**

*MSFT-1235 Deprecate old api*

- Deprecate API v 2.0.
- Update docs.

**Features**

*MSFT-1234 Car type suggester*

- Initial car type suggester implementation.
- Add insurance amount indication based on car weight.
```

13.2 Methodology

- Protect your main (DTAP) branches from direct commits. Commits shall only arrive into these branches via pull request.
- Use feature branches. Make your own prefixes (or use current ones) for classification of the ticket. Add ticket name to the name of the branch, followed by slugified ticket title.
- Prefix commits with ticket number followed by meaningful description.

Sample branch names:

- `bugfix/MSFT-1240-LinkedIn-authentication-failing`
- `deprecation/MSFT-1239-Deprecate-Python2`
- `feature/MSFT-1238-Token-authentication`

Sample commit messages:

- `MSFT-1240 Fix package configuration.`
- `MSFT-1239 Deprecate Python2.`
- `MSFT-1238 Implement token authentication.`

13.3 Release history and notes

Sequence based identifiers are used for versioning (schema follows below):

```
major.minor[.revision]
```

- It's always safe to upgrade within the same minor version (for example, from 0.3 to 0.3.4).
- Minor version changes might be backwards incompatible. Read the release notes carefully before upgrading (for example, when upgrading from 0.3.4 to 0.4).
- All backwards incompatible changes are mentioned in this document.

13.3.1 0.4.2

2019-12-25

- Minor speed-ups.
- Fix minor rendering issue with occasionally lost comments.
- Add logging.

13.3.2 0.4.1

2019-12-24

- Prevent errors and infinite wait time on faulty connections (when fetching data from Jira).
- Minor speed ups.

13.3.3 0.4

2019-12-22

- Placing `.matyan.ini` config file (placed in the home root directory now makes it a global configuration). File placed locally may override the settings. That could be handy, among others, to store credentials to Jira, which you probably do not want to have versioned.
- Make it possible to fetch additional information (for now from Jira only, but can be extended).
- Softened the regular expression patterns for ticket numbers/branch names.
- Implemented renderer classes (at the moment markdown and restructured text).
- Updated default rendering of markdown (for better markup). If you need old style behaviour, use `historical-markdown` renderer.

13.3.4 0.3.4

2019-11-27

- Add an option to show unreleased changes only.

13.3.5 0.3.3

2019-11-24

- Minor fixes.
- More tests.

13.3.6 0.3.2

2019-11-23

- Fixes in rendering logic.
- Added simple text beatification (capitalize, add final dot).

13.3.7 0.3.1

2019-11-21

- Add `headings-only` option to generate headings only (no commit messages).
- Add date to feature branch data (JSON only yet).

13.3.8 0.3

2019-11-20

- Most of the functions got optional `path` parameter to use as path to the repository directory.
- `matyan-create-config` command renamed to `matyan-make-config`.
- Next to the commands, functions are tested as well.
- Fix issue with lower edge nog being included when using dotted range.
- Added more tests.

13.3.9 0.2.1

2019-11-19

- Minor fixes.

13.3.10 0.2

2019-11-19

- Hide empty sections/records.
- Add an option to generate changelog for latest release only.
- Handle multiple merge format commit messages.
- Prevent JSON decoding errors.
- Exclude tests from coverage.

13.3.11 0.1

2019-11-18

Note: Release dedicated to my mother, who turned 70 yesterday.

- Status changed to beta.
- Minor fixes.
- Add *matyan-create-config* command.
- Add initial tests.

13.3.12 0.0.2

2019-11-17

- Minor fixes.

13.3.13 0.0.1

2019-11-17

- Initial alpha release.

CHAPTER 14

Indices and tables

- genindex
- modindex
- search