Mannou Documentation

Release 0.1.1

Muhammad Adi Prasojo

Oct 24, 2018

Contents

Mannou is a manga downloader for various sites. It can be used as *library* or *command line application*.

Note: Please remember this project still under development and created by a new programmer.

CHAPTER 1

Mannou In Action

Download your favorite manga via *command line*:

```
$ mannou https://manganelo.com/manga/aiura --download --start 2 --end 3
```

This command will download a manga called **Aiura** from chapter 2 to 3 and save it in ~/Manga/Aiura.

You can also use **Mannou** as library:

```
>>> import mannou
>>> url = 'https://manganelo.com/manga/aiura'
>>> manga = mannou.get(url)
>>> str(manga) # or manga.title
Aiura
>>> manga[0] # or manga.chapters[0]
Chapter(number='1', url='https://manganelo.com/chapter/aiura/chapter_1')
>>> images = manga.get_chapter_images(manga[0].url)
>>> images[0]
Image(name='1.jpg', url='http://s8.mkklcdn.com/mangakakalot/a1/aiura/chapter_1/1.jpg')
>>> mannou.download(url, start=1, end=5) # Download every chapters 1 until 5 in 'Aiura
↪' and save it to default location (~/Manga or %USERPROFILE%\Manga)
```

CHAPTER 2

Features

- Get manga info in-depth (using Anilist API)
- Download some or all chapters in certain manga

CHAPTER 3

Website Support

- Manganelo (English Language)
- Komikid (Bahasa Indonesia)

CHAPTER 4

The User Guide

This guide explain how you can use **Mannou**.

4.1 Introduction

Mannou is a manga downloader from various sites. You can use this project as CLI program:

```
$ mannou {url}
```

as executable module:

```
$ python3 -m mannou {url}
```

or as library:

```
>>> import mannou
```

Please read *Installation of Mannou* to install **Mannou**, or *Quickstart* if you already have it in your machine.

4.1.1 The Reason

Mannou is actually created in order to learn **Python** with actual useful software for myself. Why I choose to create a manga downloader is because I spent so much money in mobile data in order to read manga. So when I use WiFi, I can spent my time to surfing and downloading a manga in the same time. Hence, profit.

I hope this project also help you.

4.1.2 License

Free software: GNU General Public License v3.

4.2 Installation of Mannou

This part of the documentation covers the installation of **Mannou**. If you are an experienced Python developer, then just skip into *Installing Mannou* section.

4.2.1 Installing Python

But of course, you must have python installed in your machine. Verify first by typing:

```
$ python3 --version
```

if you use UNIX-like environment (Linux or MacOS), or:

```
> python --version
```

if you use Windows.

If the command is not recognized or the version is below 3.6, visit Python and follow the install instructions.

4.2.2 Installing PIP

Usually PIP already included in Python Windows installer, but not in Linux. Verify first before installing PIP by typing:

```
$ pip3 --version
```

for Linux, or:

```
> pip --version
```

for Windows.

If there is no PIP installed, please install it first by:

```
$ apt-get update
$ apt-get install python-pip3
```

On Debian-based Linux. You may need administrative privilege to install.

Note: For MacOS or other Linux distribution, please follow official guide for each vendor.

4.2.3 Installing Virtual Environment (Optional)

It is recommended to use virtual environment to separate each projects. You can use `venv`, `virtualenv`, `pipenv`, or any virtual environment you prefer.

- Venv website
- Virtualenv website
- Pipenv website

4.2.4 Installing Mannou

To install Mannou, run this in your terminal:

```
$ pip install mannou
```

And done!

4.3 Quickstart

This page provide the fastest way to getting started

First, make sure Mannou is *installed*.

Let's get started with some simple examples.

4.3.1 Getting Manga Information

Begin by importing the Mannou module:

```
>>> import mannou
```

Let's try get manga information. You can pass manga name:

```
>>> info = mannou.info('Aiura')
```

or manga url:

```
>>> info = mannou.info('https://manganelo.com/manga/aiura')
```

Now, our info is an object from `dict`, containing manga information. It is just a regular `dict`:

```
>>> info.keys()
dict_keys(['id', 'idMal', 'title', ... , 'siteUrl'])
>>> info['id']
75890
>>> info['genres']
['Comedy', 'Slice of Life']
```

Note: Passing an url instead manga's name is actually slower because Mannou need to parse first to get manga's name.

4.3.2 Parsing Manga Site

This module is useful when you want to parse a web page.

As usual, you import *mannou* first, then:

```
>>> manga = mannou.get('https://manganelo.com/manga/aiura')
```

`manga` is a *Manga* object. This object can be used to parse every images in every chapters in Aiura:

```
>>> str(manga) # or manga.title
'Aiura'
>>> manga[0] # or manga.chapters[0]
Chapter(number='1', url='https://manganelo.com/chapter/aiura/chapter_1')
```

A `manga.chapters[0]` is a `list` of *Chapter*. It is just an object of `namedtuple`. Use this information to parse images.:

```
>>> images = manga.get_chapter_images(manga[0].url)
```

`get_chapter_images` is a method for getting all images in certain chapter. In this example we want to get all images in chapter 1 of Aiura.

```
>>> images[0] # first page
Image(name='1.jpg', url='http://s8.mkklcdn.com/mangakakalot/a1/aiura/chapter_1/1.jpg')
>>> images[-1] # last page
Image(name='14.jpg', url='http://s8.mkklcdn.com/mangakakalot/a1/aiura/chapter_1/14.jpg
↪')
```

Image is an `namedtuple`, just like *Chapter*.

4.3.3 Downloading Manga

If you want to download manga, the easiest way is:

```
>>> url = 'https://manganelo.com/manga/aiura'
>>> mannou.download(url)
PosixPath('/home/<username>/Manga/Aiura')
```

This line will download every chapters in Aiura and save it in default location (`~/Manga` in Linux or `%USERPROFILE%\Manga` in Windows). It will return save location in `PosixPath` or `WindowsPath` in your machine.

If you want to download only chapter 3 to 4, use parameter `**limits`:

```
>>> mannou.download(url, start=3, end=4)
```

Maybe you want to save the manga in different location, use parameter `save_location`.:

```
>>> mannou.download(url, save_location='/home/<username>/Comic/')
PosixPath('/home/<username>/Comic/')
```

4.4 Command Line Interface

You can use Mannou as standalone program. The most basic example to use this is:

```
$ mannou https://manganelo.com/manga/aiura
```

This command will print an info about manga called *Aiura*. Id you want to download, use `--download` or `-d` flag.

```
$ mannou https://manganelo.com/manga/aiura --download
```

It will download every chapter available in Aiura and save it to your machine (default is `~/Manga/<MangaName>` or `%USERPROFILE%\Manga\MangaName`). You can change save location by using `--dest` flag.

```
$ mannou  https://manganelo.com/manga/aiura -d --dest /home/<username>/Comic/
```

To limit what chapters to download, use `--start` or `-s` and `--end` or `-e` respectively. The command below will download chapter 3 until chapter 4.

```
$ mannou  https://manganelo.com/manga/aiura -d --start 3 --end 4
```

Note: You **don't** need to remember any of those command. Just use flag `--help` or `-h` and you are good to go.

```
$ mannou --help
```

4.5 Advanced Usage

This page provide advanced usage you can do in this package.

4.5.1 Create Custom Site Parser

Do you have your favorite manga website and preferring to download manga from there, but not supported in this package? Just create your own! What you must do are:

- Make it inherit *Manga*
- Override all `abstractmethod` and have return value as same as `super().__doc__`.
- Override method `filter_chapters` if it doesn't work in your custom parser.

4.5.2 Use Custom Site Parser

Just pass your custom parser in parameter `parser` if you use main API (`info`, `get`, and `download`), or if you use *Mannou*, please see *Mannou* section.:

```
>>> from your_module import YourParser
>>> url = 'https://manganelo.com/manga/aiura'
>>> info = mannou.download(url, parser=YourParser)
```

4.5.3 AniList

Anilist is the main class for communicating with AniList API, basic usage:

```
>>> from mannou import anilist
>>> a = anilist.AniList('Aiura')
```

Now, `a` is an `Anilist` object. You can use method `json` to return JSON from AniList API.:

```
>>> a.json()
{
  'data': {
    'Media': {
      'id': 75980,
```

(continues on next page)

(continued from previous page)

```
      ...
      'siteUrl': 'https://anilist.co/manga/75980
    }
  }
}
```

Or you can use method `info` to return parsed data as :obj: *dict*.:

```
>>> a.info()
{
  'id': ['75980'],
  ...
  'siteUrl': 'https://anilist.co/manga/75980
}
```

If you are not satisfied with the default result, you can modify attribute `query` as you wish. But in order to do so, you must familiar with Anilist API and how GraphQL works.:

```
>>> a.query = """
... query ($name: String) {
...   Media (search: $name, type: MANGA) {
...     ...
...   }
... }
... """ # Your long long query
>>> a.json() # The result will be follow your query.
```

For further detail, please read this:: * AniList API Documentation * GraphQL

4.5.4 Mannou

Mannou is the main class for downloading manga. Basic usage:

```
>>> from mannou.mannou import Mannou
>>> url = 'https://manganelo.com/manga/aiura'
>>> m = Mannou(url)
```

If you have your custom parser, you can pass it in parameter `parser` directly:

```
>>> m = Mannou(url, parser=YourParser)
```

It will change value of attribute `parser` to your custom parser. You can set it like this too:

```
>>> m.parser = YourParser
```

You can also append your custom parser in attribute `parsers`. `parsers` is containing list of available parser in this package.:

```
>>> m.parsers.append(YourParser)
```

then set it automatically by calling `set_parser` method:

```
>>> m.set_parser()
```

By default, every manga will be saved it `~/Manga` or `%USERPROFILE%\Manga`. You can override it by modify `root` attribute. Please remember root attribute must be `Path` object.:

```
>>> import pathlib
>>> m.root = pathlib.Path.home().joinpath('Comic') # ~/Comic or %USERPROFILE%\\Comic
```

If preparation have already completed, download your manga by:

```
>>> m.download()
```

It will download every chapter in *https://manganelo.com/manga/aiura*. You can limit it by using parameter `start` and `end`:

```
>>> m.download(3, 7)
```

or be explicit:

```
>>> m.download(start=3, end=7)
```

It will download only chapter 3 to chapter 7.

CHAPTER 5

Source Documentation

This section provides source documentation.

5.1 mannou package

5.1.1 Subpackages

mannou.site package

Submodules

mannou.site.komikid module

Komikid parser.

class mannou.site.komikid.**Komikid**(*url*)

Bases: *mannou.parser.Manga*

Parser for http://komikid.com

For further details, please read *mannou.parser.Manga* documentation.

chapters

list of *mannou.parser.Chapter* – Available chapters.

domain = 'komikid.com'

static get_chapter_images(*chapter_url*)

Parse *chapter_url*.

Returns

- **:obj:'list' of** (obj: *mannou.parser.Image*) – List of images name and source location.
- *You can override this method as static method.*

title
str – The title of manga

mannou.site.manganelo module

Manganelo parser.

class mannou.site.manganelo.**Manganelo**(*url*)
Bases: *mannou.parser.Manga*

Parser for https://manganelo.com

For further details, please read *mannou.parser.Manga* documentation.

chapters
list of *mannou.parser.Chapter* – Available chapters.

domain = 'manganelo.com'

static get_chapter_images(*chapter_url*)
Parse *chapter_url*.

Returns

- **:obj:'list' of** (obj: *mannou.parser.Image*) – List of images name and source location.
- *You can override this method as static method.*

title
str – The title of manga

Module contents

Main parser package for parsing sites.

Every class in this package will be responsible for specific manga site.

Note: Parser class **MUST** be inherit *mannou.parser.Manga* to ensure every parser have the same functionality.

- Make sure parser class inherit *mannou.parser.Manga*.
- Implement all abstract method.
- All implemented abstract method **MUST** return. an expected object specified in *super().__doc__*.
- It strongly recommended if *number* attribute. from *mannou.parser.Chapter* is a cardinal number.
- You must override *super().filter_chapters()* if the parse do not adhere rule above.

5.1.2 Submodules

5.1.3 mannou.anilist module

Interraction module to AniList API.

See also:

mannou.api : An implementation of this module.

mannou.anilist.**API_URL = 'https://graphql.anilist.co'**
AniList GraphQL API.

mannou.anilist.**QUERY = '\nquery ($name: String) {\n Media (search: $name, type: MANGA)**
Default query that sent to AniList API.

class mannou.anilist.**AniList**(*name*)
Bases: object

Main class to communicate with AniList API.

api_url
str – AniList API url

query
str – Query that you want to sent to. It must be GraphQL query and exists in AniList API.

Parameters **name** (*str*) – Name of the manga that you want to get.

api_url = 'https://graphql.anilist.co'

info()
Get response in dictionary.

Returns Parsed response from *self.json*.

Return type dict

json()
Get response in JSON format.

Returns Response from API server in JSON format.

Return type str

query = '\nquery ($name: String) {\n Media (search: $name, type: MANGA) {\n id\n id

5.1.4 mannou.api module

Main *Mannou* API.

This module provide the easy way to use *mannou.anilist.AniList* and *mannou.mannou.Mannou*.

Note: Every functions in this module are imported in main package *mannou*. If you want to use *mannou.api.get* function, you only need to type *mannou.get*.

Please use *mannou.AniList* and *mannou.Mannou*

See also:

mannou.AniList : main class for getting manga info. *mannou.Mannou* : main class for downloading manga.

mannou.api.**info**(*search*, *parser=None*)
Get an anime info.

It can search by url or title of the manga.

Parameters **search** (*str*) – Anime title or url that you want to search.

Please remember searching by url takes longer than by name because it needs to parse an url first to get manga's title.

Returns Manga information.

Return type dict

`mannou.api.`**`get`**(*url*, *parser=None*)
Get manga chapters.

Returns

- obj: *mannou.parser.Manga* subclass. – It will return correct :obj: that handle *url* specified in *mannou.Mannou.parsers*.
- obj: *parser* – If *parser* is not None

`mannou.api.`**`download`**(*url*, *parser=None*, *save_location=None*, ***limits*)
Download chapter(s) in specified *url*.

It will download chapter(s) in and save it in your machine.

Parameters

- **`url`** (*`str`*) – URL of manga that you want to download.
- **`parser`** (*class*, optional) – Custom parser to parse *url*. It preferred that *parser* is subclassing *mannou.parser.Manga*.
- **`save_location`** (*`str, optional`*) – The save location, the default is `~/home` for UNIX or `%USERPROFILE%\Manga` for Windows.
- **`start`** (*`int, float, optional.`*) – The starting chapter, default to 0.
- **`end`** (*`int, float, optional.`*) – The last chapter that you want to download, default to None.

Returns The saved location in your machine.

Return type obj: of *pathlib.Path*

5.1.5 mannou.cli module

Main *Mannou* API.

A **command line interface** for *mannou*. This command:

```
$ mannou
```

is as same as:

```
$ python3 -m mannou
```

Example

For downloading manga from *'https://manganelo.com/manga/aiura'*, from chapter 2 to 3, you can type:

```
$ mannou https://manganelo.com/manga/aiura --start 2 --end 3
```

For further feature, please type:

```
$ mannou --help
```

mannou.cli.**main**()
Main function.

This function will run if you type:

```
$ mannou
```

in your terminal.

mannou.cli.**cli**()
Main cli function.

5.1.6 mannou.exception module

Main exception module.

Every exception must be listed in here.

exception mannou.exception.**ParserNotFoundError**
Bases: Exception

Raise when url there is no parser found in *mannou.Mannou.parsers*.

5.1.7 mannou.mannou module

Main module for downloading manga.

See also:

mannou.api : An implementation for this module.

class mannou.mannou.**Mannou**(*url*, *parser=None*)
Bases: object

Main class for *mannou*.

This class used for unify the parsers and downloading manga.

parsers
list of :class: subclassing *mannou.parser.Manga* – The stable parsers class that can parser certain site.

parser

Class subclassing *mannou.parser.Manga* – The used parser.

manga

Obj of :class: subclassing *mannou.parser.Manga* – An object that have an ability to parse *url*.

root
pathlib.Path – The save location in your machine.

Parameters

- **url** (*str*) – Manga's url.
- **parser** (*:class:, optional*) – Custom class for parsing *url*. It recommended if this class subclassing *mannou.parser.Manga*

Raises URLError – If *url* is not an url.

download(*start=0*, *end=None*)
Download manga and save it in local machine.

Parameters

- **start** (*int, float, optional*) – The first chapter, default to 0.
- **end** (*int, float, optional*) – The last chapter that you want to download, default to None.

Returns Saved manga directory if succeeded.

Return type pathlib.Path

manga = None

parse()
Assign *self.manga* to self.parser

Raises ParserNotFoundError – If *self.parser* is None

parser = None

parsers = [<class 'mannou.site.manganelo.Manganelo'>, <class 'mannou.site.komikid.Komi

root = PosixPath('/home/docs/Manga')

set_parser()
Set *self.parser* to correct parser.

5.1.8 mannou.parser module

Parser module for manga website.

This module is used for parsing manga website.

class mannou.parser.**Image**
Bases: tuple

Represent an image.

Parameters

- **name** (*str*) – The image name with extension.
- **url** (*str*) – The image source.

name
Alias for field number 0

url
Alias for field number 1

class mannou.parser.**Chapter**
Bases: tuple

Represent a chapter.

Parameters

- **number** (*str*) – The chapter number. str is used over int due there is some 'decimal' chapter, like 10.5. Why do not use float? It is weird to see chapter 1.0, I think.
- **url** (*str*) – The url of specific chapter.

number
Alias for field number 0

url
Alias for field number 1

class mannou.parser.**Manga**(*url*)
Bases: abc.ABC

An abstract base class for manga site parser.

Every parser class must inherit this class to ensure that every parser has the same functionality. The subclass also must has *domain* attribute to check whether *url* argument is valid url or not.

domain
str – Domain of the site

Parameters

- **url** (*str*) – Url of the manga.
- **soup** (BeautifulSoup) – *BeautifulSoup* object from url.

Raises URLError – If not *url* is not a valid url.

chapters
list of *mannou.parser.Chapter* – Available chapters.

classmethod check_url(*url*)
Check whether *url* is actually from *self.domain* or not.

Parameters **url** (*str*) – URL that you want to check.

Returns True if it is url from *self.domain*, False otherwise.

Return type bool

filter_chapters(*start=0*, *stop=None*)
Filter chapter in manga.

This is general algorithm for class parser that follow the rules. You may or may not override this method.

Parameters

- **start** (*int, float, optional*) – From what chapter? Default to 0.
- **stop** (*int, float, optional*) – What chapter to stop? default to None.

Returns Filtered chapters.

Return type list of *mannou.parser.Chapter*

get_chapter_images(*chapter_url*)
Parse *chapter_url*.

Returns

- **:obj:'list' of** (obj: *mannou.parser.Image*) – List of images name and source location.
- *You can override this method as static method.*

title
str – The title of manga

5.1.9 mannou.util module

Utility modules.

This module contains miscellaneous function used in every other packages and or modules.

Notes

If module grow complex, there is a chance to group similar function in the new modules.

mannou.util.**mkdir**(*dirpath*)
Create a directory.

Only create directory if `dirpath` is not exists. `dirpath` must be an object from `pathlib.Path`

Parameters **dirpath** (`pathlib.Path`) –

Returns True if successful, False otherwise.

Return type bool

mannou.util.**clear_screen**()
Clear terminal screen.

Only work for Windows and UNIX-like OS.

mannou.util.**is_url**(*url*)
Validate url.

Parameters **url** (*str*) – An url to validate.

Returns True if valid url, False otherwise.

Return type bool

mannou.util.**get_200**(*url*, *max_retries=10*, ***options*)
Sends GET request until it get 200.

By default, it will try 10 times before raise.

Parameters

- **url** (*str*) – URL that you want to GET
- **max_retries** (*int, optional*) – Decide how many times sending GET request before raise.
- **user_agent** (*str, optional*) – User agent that you want to use
- **stream** (*bool, optional*) – Decide if you want to stream or not.

Returns

Return type `requests.Response`

Raises `HTTPError` – If *max_retries* exceeded.

mannou.util.**make_soup**(*url*)
Create *bs4.BeautifulSoup* object from url.

Parameters **url** (*str*) – URL that you want to scrap.

Returns

Return type `bs4.BeautifulSoup`

Raises exception.HTTPError – From *mannou.util.get_200*

mannou.util.**download**(*url*, *filepath*)
Downloader, with progress bar.

Send GET request and save it to local computer.

Parameters

- **url** (*str*) – URL that you want to download.
- **filepath** (str) – Saved file location

class mannou.util.**StatusCode**
Bases: object

A bunch of HTTP status codes.

Every HTTP status code stored in readable attributes name.

INTERNAL_SERVER_ERROR = 500

NOT_FOUND = 404

OK = 200

5.1.10 Module contents

Mannou Manga Downloader

Mannou is a manga downloader for downloading manga from various sites. Basic usage:

```
>>> import mannou
>>> url = 'https://manganelo.com/manga/aiura'
>>> manga = mannou.get(url)
>>> str(manga) # or manga.title
Aiura
>>> manga[0] # or manga.chapters[0]
Chapter(number='1', url='https://manganelo.com/chapter/aiura/chapter_1')
>>> images = manga.get_chapter_images(manga[0].url)
>>> images[0]
Image(name='1.jpg', url='http://s8.mkklcdn.com/mangakakalot/a1/aiura/chapter_1/1.jpg')
>>> mannou.download(url, start=1, end=5) # Download every chapters 1 until 5 in 'Aiura
↪' and save it to default location (~/Manga or %USERPROFILE%\Manga)
```

The other methods and properties are supported. Full documentation is at <https://mannou.readthedocs.io>.

5.2 mannou

Python Module Index

m

Index

P

Q

R

S

T

U