Metadata and Docs URLs

You can customize several metadata configurations in your **FastAPI** application.

Metadata for API

You can set the following fields that are used in the OpenAPI specification and the automatic API docs UIs:

Parameter	Type	Description
`title`	`str`	The title of the API.
`description`	`str`	A short description of the API. It can use Markdown.
`version`	`string`	The version of the API. This is the version of your own application, not of OpenAPI. For example `2.5.0`.
`terms_of_service`	`str`	A URL to the Terms of Service for the API. If provided, this has to be a URL.
`contact`	`dict`	The contact information for the exposed API. It can contain several fields. ▶ `contact` fields
`license_info`	`dict`	The license information for the exposed API. It can contain several fields. ▶ `license_info` fields

You can set them as follows:

```
{!../../../docs_src/metadata/tutorial001.py!}
```

!!! tip You can write Markdown in the `description` field and it will be rendered in the output.

With this configuration, the automatic API docs would look like:

Metadata for tags

You can also add additional metadata for the different tags used to group your path operations with the parameter `openapi_tags`.

It takes a list containing one dictionary for each tag.

Each dictionary can contain:

- `name` (**required**): a `str` with the same tag name you use in the `tags` parameter in your *path operations* and `APIRouter` s.
- `description` : a `str` with a short description for the tag. It can have Markdown and will be shown in the docs UI.
- `externalDocs` : a `dict` describing external documentation with:
    - `description` : a `str` with a short description for the external docs.
    - `url` (**required**): a `str` with the URL for the external documentation.

Create metadata for tags

Let's try that in an example with tags for `users` and `items`.

Create metadata for your tags and pass it to the `openapi_tags` parameter:

```
{!../../../docs_src/metadata/tutorial004.py!}
```

Notice that you can use Markdown inside of the descriptions, for example "login" will be shown in bold (**login**) and "fancy" will be shown in italics (*fancy*).

!!! tip You don't have to add metadata for all the tags that you use.

Use your tags

Use the `tags` parameter with your *path operations* (and `APIRouter`s) to assign them to different tags:

```
{!../../../docs_src/metadata/tutorial004.py!}
```

!!! info Read more about tags in Path Operation Configuration{.internal-link target=_blank}.

Check the docs

Now, if you check the docs, they will show all the additional metadata:

Order of tags

The order of each tag metadata dictionary also defines the order shown in the docs UI.

For example, even though `users` would go after `items` in alphabetical order, it is shown before them, because we added their metadata as the first dictionary in the list.

OpenAPI URL

By default, the OpenAPI schema is served at `/openapi.json`.

But you can configure it with the parameter `openapi_url`.

For example, to set it to be served at `/api/v1/openapi.json`:

```
{!../../../docs_src/metadata/tutorial002.py!}
```

If you want to disable the OpenAPI schema completely you can set `openapi_url=None`, that will also disable the documentation user interfaces that use it.

Docs URLs

You can configure the two documentation user interfaces included:

- **Swagger UI**: served at `/docs`.

    - You can set its URL with the parameter `docs_url`.
    - You can disable it by setting `docs_url=None`.
- ReDoc: served at `/redoc`.
    - You can set its URL with the parameter `redoc_url`.
    - You can disable it by setting `redoc_url=None`.

For example, to set Swagger UI to be served at `/documentation` and disable ReDoc:

```
{!../../../docs_src/metadata/tutorial003.py!}
```