# Итоговое задание

GitHub: https://github.com/uaaaaau/SEB-Final-Task/tree/main
Docker Hub: https://hub.docker.com/repositories/aaauuuu

## TODO-сервис

Создание образа, загрузка на Docker Hub и запуск контейнера:

```
/home/mobaxterm/final_project/todo_app
2025-01-05    14:43.18    docker build -t aaauuuu/todo-service:latest .

[+] Building 3.1s (11/11) FINISHED                            docker:desktop-linux
 => [internal] load build definition from Dockerfile                        0.0s
 => => transferring dockerfile: 337B                                        0.0s
 => [internal] load metadata for docker.io/library/python:3.9-slim          1.9s
 => [auth] library/python:pull token for registry-1.docker.io               0.0s
 => [internal] load .dockerignore                                           0.0s
 => => transferring context: 2B                                             0.0s
 => [1/5] FROM docker.io/library/python:3.9-slim@sha256:caaf1af9e23adc614   0.0s
 => => resolve docker.io/library/python:3.9-slim@sha256:caaf1af9e23adc614   0.0s
 => [internal] load build context                                           0.0s
 => => transferring context: 154B                                           0.0s
 => CACHED [2/5] WORKDIR /app                                               0.0s
 => CACHED [3/5] COPY requirements.txt .                                    0.0s
 => CACHED [4/5] RUN pip install --no-cache-dir -r requirements.txt         0.0s
 => CACHED [5/5] COPY . .                                                   0.0s
 => exporting to image                                                      0.9s
 => => exporting layers                                                     0.0s
 => => exporting manifest sha256:4a137db27495375ee3fc8e82fe2f19d0b19abe13   0.0s
 => => exporting config sha256:569e9831b34c297f36dc25202a0e1ea51cbc56c397   0.0s
 => => exporting attestation manifest sha256:cf55df30a03e09010bb79093cbd4   0.0s
 => => exporting manifest list sha256:711f131999e43958d7fcd3fb2b26990d40b   0.0s
 => => naming to docker.io/aaauuuu/todo-service:latest                      0.0s
 => => unpacking to docker.io/aaauuuu/todo-service:latest                   0.8s
                                                                              ✔
```

```
/home/mobaxterm/final_project/todo_app
2025-01-05    14:44.12    docker push aaauuuu/todo-service:latest
The push refers to repository [docker.io/aaauuuu/todo-service]
16c49ccbe2da: Pushed
75a2bc32319e: Pushed
11bf94da8b1c: Pushed
9feece89654b: Pushed
d381bf0bfd6e: Pushed
15ae4a6da7c5: Pushed
b6fbd1f29b29: Pushed
fd674058ff8f: Pushed
96df0e5e8179: Pushed
latest: digest: sha256:711f131999e43958d7fcd3fb2b26990d40ba4150ada941c237d271664
260f23a size: 856
                                                                              ✔
```

```
/home/mobaxterm/final_project/todo_app
2025-01-05    14:46.09    docker run -d -p 8000:80 -v todo_data:/app/data a
aauuuu/todo-service:latest
1b62a4da06380bb2f245822ba5c66425358c75fc50c8d8d1056b9c7c44a223df
                                                                              ✔
```

## Интерфейс приложения TODO-сервиса:



## Создание задач:

**Responses**

Curl

```
curl -X 'POST' \
  'http://localhost:8000/items' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "title": "Watch TV",
  "description": "Watch sports on TV",
  "completed": false
}'
```

Request URL

```
http://localhost:8000/items
```

Server response

| Code | Details |
|------|---------|
| 200  | Response body |

```
{
  "id": 1,
  "title": "Watch TV",
  "description": "Watch sports on TV",
  "completed": false
}
```

Response headers

```
content-length: 80
content-type: application/json
date: Sun,05 Jan 2025 12:05:27 GMT
server: uvicorn
```

---

**POST** /items Create Item

**Parameters**

Cancel   Reset

No parameters

**Request body** required

application/json ▾

```
{
  "title": "Clean the house",
  "description": "Clean the dust and vacuum the carpets",
  "completed": false
}
```

Execute   Clear

---

**Responses**

Curl

```
curl -X 'POST' \
  'http://localhost:8000/items' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "title": "Clean the house",
  "description": "Clean the dust and vacuum the carpets",
  "completed": false
}'
```

Request URL

```
http://localhost:8000/items
```

Server response

| Code | Details |
|------|---------|
| 200  | Response body |

```
{
  "id": 2,
  "title": "Clean the house",
  "description": "Clean the dust and vacuum the carpets",
  "completed": false
}
```

Response headers

```
content-length: 106
content-type: application/json
date: Sun,05 Jan 2025 12:08:05 GMT
server: uvicorn
```

**POST** `/items` Create Item                                                                      ⌃

## Parameters                                                          Cancel          Reset

No parameters

**Request body** required                                                          application/json  ⌄

```
{
  "title": "Buy groceries",
  "description": "Bread, butter",
  "completed": true
}
```

| Execute | Clear |
|---------|-------|

## Responses

Curl

```
curl -X 'POST' \
  'http://localhost:8000/items' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "title": "Buy groceries",
  "description": "Bread, butter",
  "completed": true
}'
```

Request URL

```
http://localhost:8000/items
```

Server response

| Code | Details |
|------|---------|
| 200  | Response body |

```
{
  "id": 3,
  "title": "Buy groceries",
  "description": "Bread, butter",
  "completed": true
}
```

Response headers

```
content-length: 79
content-type: application/json
date: Sun,05 Jan 2025 12:09:24 GMT
server: uvicorn
```

## Получение списка всех дел:

GET **/items** Get Items

**Parameters**
Cancel

No parameters

| Execute | Clear |

**Responses**

Curl

```
curl -X 'GET' \
  'http://localhost:8000/items' \
  -H 'accept: application/json'
```

Request URL

```
http://localhost:8000/items
```

Server response

| Code | Details |
|------|---------|
| 200 | Response body |

```
[
  {
    "id": 1,
    "title": "Watch TV",
    "description": "Watch sports on TV",
    "completed": false
  },
  {
    "id": 2,
    "title": "Clean the house",
    "description": "Clean the dust and vacuum the carpets",
    "completed": false
  },
  {
    "id": 3,
    "title": "Buy groceries",
    "description": "Bread, butter",
    "completed": true
  }
]
```

Download

Response headers

```
content-length: 269
content-type: application/json
date: Sun,05 Jan 2025 12:10:28 GMT
server: uvicorn
```

## Получение элемента по id:

GET **/items/{item_id}** Get Item

**Parameters**
Cancel

| Name | Description |
|------|-------------|
| **item_id** * required <br> integer <br> (path) | 2 |

| Execute | Clear |

**Responses**

Curl

```
curl -X 'GET' \
  'http://localhost:8000/items/2' \
  -H 'accept: application/json'
```

Request URL

```
http://localhost:8000/items/2
```

Server response

| Code | Details |
|------|---------|
| 200  | Response body |

```
{
  "id": 2,
  "title": "Clean the house",
  "description": "Clean the dust and vacuum the carpets",
  "completed": false
}
```

Download

Response headers

```
content-length: 106
content-type: application/json
date: Sun,05 Jan 2025 12:12:23 GMT
server: uvicorn
```

Редактирование элемента:

**PUT** /items/{item_id} Update Item

Parameters

Cancel | Reset

| Name | Description |
|------|-------------|
| item_id * required<br>integer<br>(path) | 2 |

Request body required                                application/json

```
{
  "title": "Clean the house",
  "description": "Clean the dust and vacuum the carpets",
  "completed": true
}
```

Execute | Clear

Curl

```
curl -X 'PUT' \
  'http://localhost:8000/items/2' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "title": "Clean the house",
  "description": "Clean the dust and vacuum the carpets",
  "completed": true
}'
```

Request URL

```
http://localhost:8000/items/2
```

Server response

| Code | Details |
| --- | --- |
| 200 | Response body |

```
{
  "id": 2,
  "title": "Clean the house",
  "description": "Clean the dust and vacuum the carpets",
  "completed": true
}
```

Response headers

```
content-length: 105
content-type: application/json
date: Sun,05 Jan 2025 12:14:29 GMT
server: uvicorn
```

Удаление элемента:

**DELETE** /items/{item_id} Delete Item

Parameters                                                                                    Cancel

| Name | Description |
| --- | --- |
| item_id * required<br>integer<br>(path) | 2 |

| Execute | Clear |
| --- | --- |

Responses

Curl

```
curl -X 'DELETE' \
  'http://localhost:8000/items/2' \
  -H 'accept: application/json'
```

Request URL

```
http://localhost:8000/items/2
```

Server response

| Code | Details |
| --- | --- |
| 200 | Response body |

```
{
  "message": "Item deleted"
}
```

Response headers

```
content-length: 26
content-type: application/json
date: Sun,05 Jan 2025 12:22:20 GMT
server: uvicorn
```

Проверка удаления:

GET  /items  Get Items

Parameters                                                              Cancel

No parameters

                    Execute                    |                    Clear

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8000/items' \
  -H 'accept: application/json'
```

Request URL

```
http://localhost:8000/items
```

Server response

| Code | Details |
|------|---------|
| 200  | Response body |

```
[
  {
    "id": 1,
    "title": "Watch TV",
    "description": "Watch sports on TV",
    "completed": false
  },
  {
    "id": 3,
    "title": "Buy groceries",
    "description": "Bread, butter",
    "completed": true
  }
]
```

Response headers

```
content-length: 162
content-type: application/json
date: Sun,05 Jan 2025 12:22:27 GMT
server: uvicorn
```

# ShortURL-сервис

Создание образа, загрузка на Docker Hub и запуск контейнера:

```
/home/mobaxterm/final_project/shorturl_app
2025-01-05   15:16.37   docker build -t aaauuuu/shorturl-service:latest .

[+] Building 2.7s (11/11) FINISHED                          docker:desktop-linux
 => [internal] load build definition from Dockerfile                       0.0s
 => => transferring dockerfile: 347B                                       0.0s
 => [internal] load metadata for docker.io/library/python:3.9-slim         2.1s
 => [auth] library/python:pull token for registry-1.docker.io             0.0s
 => [internal] load .dockerignore                                          0.0s
 => => transferring context: 2B                                            0.0s
 => [1/5] FROM docker.io/library/python:3.9-slim@sha256:caaf1af9e23adc614  0.0s
 => => resolve docker.io/library/python:3.9-slim@sha256:caaf1af9e23adc614  0.0s
 => [internal] load build context                                          0.0s
 => => transferring context: 3.29kB                                        0.0s
 => CACHED [2/5] WORKDIR /app                                              0.0s
 => CACHED [3/5] COPY requirements.txt .                                   0.0s
 => CACHED [4/5] RUN pip install --no-cache-dir -r requirements.txt        0.0s
 => [5/5] COPY . .                                                         0.0s
 => exporting to image                                                     0.2s
 => => exporting layers                                                    0.1s
 => => exporting manifest sha256:5702db3143bf00cd7de4edcba729c537bd35f86e  0.0s
 => => exporting config sha256:d9cb0d5cdfcc6bd724c7a59fa335cbed8d488fc217  0.0s
 => => exporting attestation manifest sha256:b0de735ca85b3c27a9b1329c092d  0.0s
 => => exporting manifest list sha256:69b308648a3997e853cf89a0ca1a97c9394  0.0s
 => => naming to docker.io/aaauuuu/shorturl-service:latest                 0.0s
 => => unpacking to docker.io/aaauuuu/shorturl-service:latest              0.0s
                                                                              ✓

/home/mobaxterm/final_project/shorturl_app
2025-01-05   15:25.10   docker push aaauuuu/shorturl-service:latest
The push refers to repository [docker.io/aaauuuu/shorturl-service]
b6fbd1f29b29: Pushed
f97e228b313d: Pushed
3770894383c4: Pushed
d381bf0bfd6e: Pushed
11bf94da8b1c: Pushed
9feece89654b: Pushed
fd674058ff8f: Pushed
75a2bc32319e: Pushed
96df0e5e8179: Pushed
latest: digest: sha256:69b308648a3997e853cf89a0ca1a97c9394cde68537c716e59083ca49
1d12f16 size: 856
                                                                              ✓

/home/mobaxterm/final_project/shorturl_app
2025-01-05   15:25.57   docker run -d -p 8000:80 -v shorturl_data:/app/da
ta aaauuuu/shorturl-service:latest
6220143b7e6ea07631dac31defd556233560184b083e9784a637106b57f67fdc
```
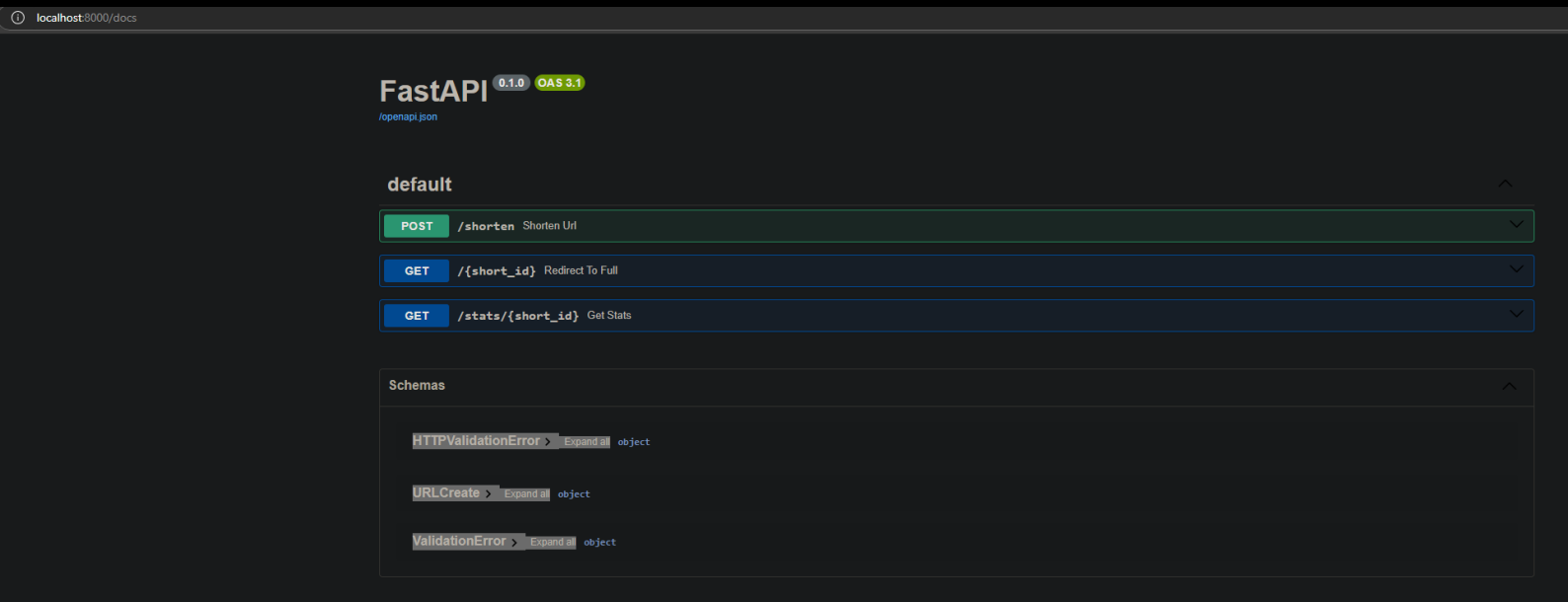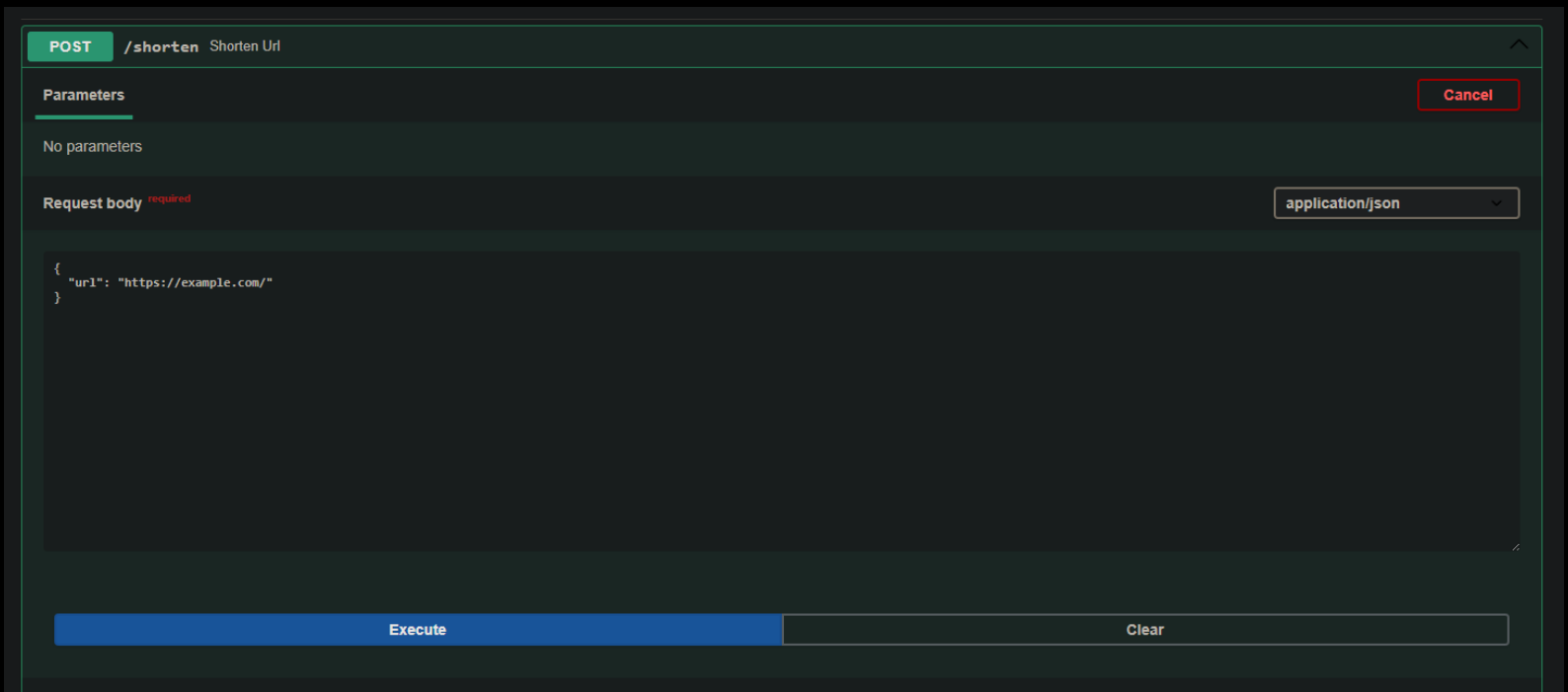
Интерфейс сервиса для сокращения ссылок:

# FastAPI 0.1.0 OAS 3.1

/openapi.json

## default

| POST | /shorten | Shorten Url |

| GET | /{short_id} | Redirect To Full |

| GET | /stats/{short_id} | Get Stats |

### Schemas

HTTPValidationError › Expand all   object

URLCreate › Expand all   object

ValidationError › Expand all   object

Сокращение URL:

| POST | /shorten | Shorten Url |

**Parameters**                                                    Cancel

No parameters

**Request body** required                                      application/json

```
{
  "url": "https://example.com/"
}
```

Execute                                              Clear

**Responses**

**Curl**

```
curl -X 'POST' \
  'http://localhost:8000/shorten' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "url": "https://example.com/"
}'
```

**Request URL**

```
http://localhost:8000/shorten
```

**Server response**

| Code | Details |
| --- | --- |
| 200 | **Response body** |

```
{
  "short_url": "http://localhost:8000/o46Yh6"
}
```

**Response headers**

```
content-length: 44
content-type: application/json
date: Sun,05 Jan 2025 12:42:58 GMT
server: uvicorn
```

Переход на страницу по сокращенной ссылке:



https://example.com

**Example Domain**

This domain is for use in illustrative examples in documents. You may use this domain in literature without prior coordination or asking for permission.

More information...

Вывод данных о ссылке:



Образы сервисов загружены на DockerHub: