

TaskTracker API (1.0)

документация

Навигация:

- [Авторизация и аутентификация](#)
- [Пользователи](#)
- [Таски](#)
- [Статусы](#)
- [Комментарии](#)

Тэги в документации:

- #Авторизация - для доступа к эндпоинту нужно передать заголовок с авторизацией
- #Без_авторизации - для доступа к эндпоинту не требуется заголовок с авторизацией
- #Пагинация - эндпоинт использует пагинацию в ответе
- #Суперпользователь - для доступа к эндпоинту нужны права супер-пользователя

Авторизация и аутентификация (Authorization and Authentication)

Авторизация проекта построена на PyJWT Bearer Token

Вход пользователя

Вход пользователя осуществляется через эндпоинт `POST: /api/v1/auth/login/`

Тело запроса:

```
{
  "email": "user@example.com",
  "password": "string"
}
```

На вход принимается почта пользователя (`email`) и пароль (`password`)

Успешный ответ:

```
{
  "access_token": "string",
  "refresh_token": "string",
  "token_type": "Bearer"
}
```

Успешный ответ содержит access-токен, refresh-токен и его тип (`token_type`), который следует использовать для авторизации пользователя.

Регистрация пользователя

Регистрация пользователя осуществляется через эндпоинт `POST:`

`/api/v1/auth/register/`

Тело запроса:

```
{
  "username": "string",
  "first_name": "string",
  "last_name": "string",
  "email": "user@example.com",
  "password": "string"
}
```

В теле запроса обязательны следующие параметры:

Юзернейм пользователя - `username`

Имя - `first_name`

Фамилия - `last_name`

Почта - `email`

Пароль - `password`

При успешном создании пользователя сервер вернет следующий ответ:

```
{
  "username": "string",
  "first_name": "string",
  "last_name": "string",
  "id": 0,
  "full_name": "string",
  "avatar": "string",
  "about": "string"
}
```

Передача токена для авторизации пользователя

- Авторизация пользователя осуществляется через заголовок `Authorization`. Для авторизации передайте тип токена и сам токен в заголовок.

Пример: `Authorization: Bearer MyToken`

Обновление access-токена через refresh-токен

- Обновление access-токена осуществляется через endpoint `api/v1/auth/refresh/`

В заголовке авторизации нужно указать refresh-токен полученный при входе.

Пагинация в проекте

Пример пагинации:

```
{
  "results": [
    {
      "username": "bob",
      "first_name": "Bob",
      "last_name": "Mayakov",
      "id": 1,
      "full_name": "Bob Mayakov",
      "avatar": "string",
      "about": "string"
    },
  ],
  "page": 1,
  "pages_count": 1,
  "limit": 100
}
```

Эндпоинты содержащие пагинацию могут принимать дополнительные параметры в запросе:

page - номер страницы (по умолчанию 1)

limit - ограничение объектов на странице (по умолчанию 100)

Пример запроса с дополнительными параметрами:

```
api/v1/users/?page=7&limit=20
```

Users (Пользователи)

Получения списка всех пользователей

- #Пагинация
- #Без_авторизации

GET: api/v1/users/

Успешный ответ:

```
{
  "results": [
    {
      "username": "string",
      "first_name": "string",
      "last_name": "string",
      "id": 1,
      "full_name": "string",
      "avatar": "string",
      "about": "string"
    }
  ],
  "page": 0,
  "pages_count": 0,
  "limit": 0
}
```

Получение одного пользователя по имени

- #Без_авторизации

GET: api/v1/users/@username/

Успешный ответ:

```
{
  "username": "string",
  "first_name": "string",
  "last_name": "string",
  "id": 0,
  "full_name": "string",
  "avatar": "string",
  "about": "string"
}
```

Получение профиля авторизованного пользователя

- #Авторизация

GET: api/v1/users/me/

Успешный ответ:

```
{
  "username": "string",
  "first_name": "string",
  "last_name": "string",
  "id": 0,
  "full_name": "string",
  "avatar": "string",
  "about": "string",
  "email": "user@example.com",
  "created_at": "2024-07-30T03:08:34.735Z",
  "updated_at": "2024-07-30T03:08:34.735Z",
  "is_active": bool
}
```

Обновление авторизованного пользователя

- #Авторизация

Доступна частичная передача параметров при обновлении

PATCH: api/v1/users/me/

Тело запроса:

```
{
  "username": "string",
  "first_name": "string",
  "last_name": "string",
  "email": "user@example.com",
  "about": "string"
}
```

Успешный ответ:

```
{
  "username": "string",
  "first_name": "string",
  "last_name": "string",
  "id": 0,
  "full_name": "string",
  "avatar": "string",
  "about": "string"
}
```

Обновление аватара пользователя

- #Авторизация

PATCH: `api/v1/users/me/update-avatar/`

Тело запроса:

Изображение

Успешный ответ:

```
{
  "username": "string",
  "first_name": "string",
  "last_name": "string",
  "id": 1,
  "full_name": "string",
  "avatar": "string",
  "about": "string"
}
```

Tasks (Таски)

Создание таска

- #Авторизация

POST: `api/v1/tasks/`

Тело запроса:

```
{
  "title": "string",
  "description": "string",
  "marked_users": [
    "string"
  ]
}
```

В `marked_users` передаются юзернеймы пользователей, которые будут отмечены при создании задачи

Успешный ответ:

```
{
  "title": "string",
  "description": "string",
  "id": 1,
  "author": {
    "username": "string",
    "first_name": "string",
    "last_name": "string",
    "id": 2,
    "full_name": "string",
    "avatar": "string",
    "about": "string"
  },
  "status": {
    "name": "string",
    "slug": "string",
    "id": 1
  },
  "marked_users": [
    {
      "username": "string",
      "first_name": "string",
      "last_name": "string",
      "id": 1,
      "full_name": "string",
      "avatar": "string",
      "about": "string"
    }
  ],
  "created_at": "string",
  "updated_at": "string"
}
```

Получение задач авторизованного пользователя

- #Авторизация
- #Пагинация

GET: api/v1/tasks/

Успешный ответ:

```
{
  "results": [
    {
      "title": "string",
      "description": "string",
      "id": 1,
      "author": {
        "username": "string",
        "first_name": "string",
        "last_name": "string",
        "id": 1,
        "full_name": "string",
        "avatar": "string",
        "about": "string"
      },
      "status": {
        "name": "string",
        "slug": "string",
        "id": 1
      },
      "marked_users": [
        {
          "username": "string",
          "first_name": "string",
          "last_name": "string",
          "id": 1,
          "full_name": "string",
          "avatar": "string",
          "about": "string"
        },
        {
          "username": "string",
          "first_name": "string",
          "last_name": "string",
          "id": 1,
          "full_name": "string",
          "avatar": "string",
          "about": "string"
        }
      ]
    }
  ]
}
```



```
    ],
    "created_at": "string",
    "updated_at": "string"
  },
],
"page": 1,
"pages_count": 1,
"limit": 20
}
```

Получение списка задач где отмечен авторизованный пользователь

- #Авторизация

GET: `api/v1/tasks/marked-in-tasks/`

Успешный ответ:

```
{
  "results": [
    {
      "title": "string",
      "description": "string",
      "id": 1,
      "author": {
        "username": "string",
        "first_name": "string",
        "last_name": "string",
        "id": 2,
        "full_name": "string",
        "avatar": "string",
        "about": "string"
      },
      "status": {
        "name": "string",
        "slug": "string",
        "id": 1
      },
      "marked_users": [
        {
          "username": "string",
          "first_name": "string",
          "last_name": "string",
          "id": 1,
          "full_name": "string",

```

```
        "avatar": "string",
        "about": "string"
      }
    ],
    "created_at": "string",
    "updated_at": "string"
  }
],
"page": 0,
"pages_count": 0,
"limit": 0
}
```

Получение одного таска

- #Авторизация

Получить таск может только автор таска или отмеченный в нем пользователь

GET: `api/v1/tasks/{task_id}/`

Успешный ответ:

```
{
  "title": "string",
  "description": "string",
  "id": 1,
  "author": {
    "username": "string",
    "first_name": "string",
    "last_name": "string",
    "id": 2,
    "full_name": "string",
    "avatar": "string",
    "about": "string"
  },
  "status": {
    "name": "string",
    "slug": "string",
    "id": 1
  },
  "marked_users": [
    {
      "username": "string",
      "first_name": "string",
      "last_name": "string",

```

```

        "id": 1,
        "full_name": "string",
        "avatar": "string",
        "about": "string"
    }
],
"created_at": "string",
"updated_at": "string",
"comments": [
    {
        "text": "string",
        "id": 1,
        "author": {
            "username": "string",
            "first_name": "string",
            "last_name": "string",
            "id": 1,
            "full_name": "string",
            "avatar": "string",
            "about": "string"
        },
        "created_at": "string"
    }
]
}

```

Обновление задачи

- #Авторизация

PUT: `api/v1/tasks/{task_id}/`

Тело запроса:

```

{
    "title": "string",
    "description": "string",
    "marked_users": [
        "string"
    ],
    "status_id": 0
}

```

В `marked_users` передаются все пользователи, которые будут отмечены в задаче после обновления

Успешный ответ:

```
{
  "title": "string",
  "description": "string",
  "id": 0,
  "author": {
    "username": "string",
    "first_name": "string",
    "last_name": "string",
    "id": 0,
    "full_name": "string",
    "avatar": "string",
    "about": "string"
  },
  "status": {
    "name": "string",
    "slug": "string",
    "id": 0
  },
  "marked_users": [
    {
      "username": "string",
      "first_name": "string",
      "last_name": "string",
      "id": 0,
      "full_name": "string",
      "avatar": "string",
      "about": "string"
    }
  ],
  "created_at": "string",
  "updated_at": "string"
}
```

Удаление задачи

- #Авторизация

DELETE: `api/v1/tasks/{task_id}/`

Успешный ответ:

204 NO CONTENT

Task-Statuses (Статусы Задачи)

Создание, обновление и удаление статусов доступно только супер-пользователям

Получение списка всех статусов

- #Без_авторизации

GET: api/v1/task-statuses/

Успешный ответ:

```
[
  {
    "name": "string",
    "slug": "string",
    "id": 0
  }
]
```

Создание статуса

- #Суперпользователь

POST: api/v1/task-statuses/

Тело запроса:

```
{
  "name": "string",
  "slug": "string"
}
```

Успешный ответ:

```
{
  "name": "string",
  "slug": "string",
  "id": 0
}
```

Обновление статуса

- #Суперпользователь

```
PATCH: api/v1/task-statuses/{status_id}/
```

Тело запроса:

```
{
  "name": "string",
  "slug": "string"
}
```

Успешный ответ:

```
{
  "name": "string",
  "slug": "string",
  "id": 0
}
```

Удаление статуса

- #Суперпользователь

```
DELETE: api/v1/task-statuses/{status_id}/
```

Успешный ответ:

```
204 NO CONTENT
```

Task-Comments (Комментарии Тасков)

Создание комментария

- #Авторизация

```
POST: api/v1/task-comments/{task_id}/
```

Тело запроса:

```
{
  "text": "string"
}
```

Успешный ответ:

```
201 CREATED
```

Получение комментария

- #Авторизация

```
GET: api/v1/task-comments/{comment_id}/
```

Успешный ответ:

```
{
  "text": "string",
  "id": 38,
  "author": {
    "username": "string",
    "first_name": "string",
    "last_name": "string",
    "id": 7,
    "full_name": "string",
    "avatar": "string",
    "about": "string"
  },
  "created_at": "string"
}
```

Обновление комментария

- #Авторизация

```
PATCH: api/v1/task-comments/{comment_id}/
```

Тело запроса:

```
{
  "text": "string"
}
```

Успешный ответ:

200 OK

Удаление комментария

- #Авторизация

```
DELETE: api/v1/task-comments/{comment_id}/
```

Успешный ответ:

204 NO CONTENT