



Описание API

Базовый адрес API: <http://<servername>/api>

Конечные точки API:

Регистрация/создание нового пользователя	3
Получить данные для JWT аутентификации.....	4
Обновить данные для JWT аутентификации	5
Создать организацию	6
Получить список организаций	8
Получить пользователя по id	10
Редактирование пользователя по id.....	12
Получить список пользователей	14

Регистрация/создание нового пользователя: [/signup](#)**Метод: `post`****headers:**

```
{  
  'Content-Type': 'application/json',  
}
```

json:

```
{  
  'email': <email>,  
  'password': <password>,  
  'password2': <password>  
}
```

response:

В случае успешной регистрации:

```
{ 'message': 'Registration success' }
```

В случае ошибочных данных:

```
{ 'error': 'Such user is exist' }
```

Получить данные для JWT аутентификации: `/token`

Метод: **post**

headers:

```
{  
  'Content-Type': 'application/json',  
}
```

json:

```
{  
  'email': <email>,  
  'password': <password>,  
}
```

response:

```
{  
  'access': <JWT>,  
  'refresh': <JWT to refresh>,  
}
```

access – JWT ключ для аутентификации запросов к API;

refresh – JWT ключ для обновления ключа **access** по истечении его срока (срок ключа **access** – 240 часов)

Обновить данные для JWT аутентификации: [/token/refresh](#)**Метод: `post`****headers:**

```
{  
  'Content-Type': 'application/json',  
}
```

json:

```
{  
  'refresh': <JWT to refresh>,  
}
```

<JWT to refresh> – значение, полученное в ключе **refresh** json-ответа при **post**-запросе на конечную точку [/token](#)

response:

```
{  
  'access': <JWT>,  
}
```

Создать организацию: [/organization/create](#)**Метод: `post`****headers:**

```
{  
  'Authorization': 'Bearer <JWT>',  
  'Content-Type': 'application/json'  
}
```

json:

```
{  
  'name': '<Название организации>',  
  'description': '<Описание организации>',  
  'users': <Список id пользователей [ 1, ..., n ]>  
}
```

***name** – обязательное поле

response:

```
{  
  'id': <id организации>,  
  'users': <Список email-ов пользователей организации:  
    [ "user-email-1@mail.ru",  
      ...,  
      "user-email-n@mail.ru" ]> ,  
  'name': '<Название организации>',  
  'description': 'Описание организации'  
}
```

Получить список организаций: [/organizations](#)**Метод: `get`****headers:**

```
{  
    'Authorization': 'Bearer <JWT>',  
    'Content-Type': 'application/json'  
}
```

params:

```
{  
    'page': <Номер страницы>,  
    'page_size': <Число объектов на одной странице>  
}
```


response:

```
{
  'organizations': [
    {
      'id': <id организации-1>,
      'users': <Список пользователей [
        {
          'id': <id пользователя-1>,
          'email': '<email пользователя-1>',
          'phone': '<Телефонный номер пользователя-1>',
          'first_name': '<Имя пользователя-1>',
          'last_name': '<Фамилия пользователя-1>',
          'avatar': '<Путь к аватару пользователя-1>',
          'organization_set': <Список организаций пользователя-1
            [ 'Организация-1', ..., 'Организация-n' ]
          },
        ...
      ]>,
      'name': '<Имя организации-1>',
      'description': '<Описание организации-1>'
    },
    ...
  ],
  'previous_url': '<Ссылка на предыдущую страницу>',
  'next_url': '<Ссылка на следующую страницу>',
  'page_links': <Список ссылок на страницы [
    [ '<url>', <Номер страницы>, <Активная?: bool>, <Рабочая?: bool> ],
    ...,
  ]>
}
```

Получить пользователя по id: `/user/?pk=id`

Метод: `get`

headers:

```
{  
  'Authorization': 'Bearer <JWT>',  
  'Content-Type': 'application/json'  
}
```

params:

```
{  
  'pk': <id пользователя>  
}
```

response:

```
{
  'id': <id пользователя>,
  'email': '<email-пользователя@mail.ru>',
  'phone': '<Телефонный номер пользователя>',
  'first_name': '<Имя пользователя>',
  'last_name': '<Фамилия пользователя>',
  'avatar': '<http://абсолютная/ссылка/на/файл/аватара.ext>',
  'organization_set': <Список организаций [
    {
      'id': <id организации>,
      'users': <Список пользователей организации-1:
        [ "email-пользователя-1 ",
          ...,
          "email-пользователя-n" ]>,
      'name': '<Название организации>',
      'description': '<Описание организации>'
    },
    ...
  ]>
}
```

Редактирование пользователя по id: /user/edit/?pk=id**Метод:** put**headers:**

```
{  
    'Authorization': 'Bearer <JWT>',  
    'Content-Type': 'application/json'  
}
```

json: {

```
    'id': <id пользователя>,  
    'email': '<email-пользователя@mail.ru>',  
    'phone': '<Телефонный номер пользователя>',  
    'first_name': '<Имя пользователя>',  
    'last_name': '<Фамилия пользователя>',  
    'avatar': '<http://абсолютная/ссылка/на/файл/аватара.ext>',  
    'organization_set'*: <Список id организаций [ 1, ..., n ]  
}
```

params:

```
{  
    'pk': <id пользователя>  
}
```

* **!NB:** Список, переданный в 'organization_set' заменит прежний

response:

```
{
  'id': <id пользователя>,
  'email': '<email-пользователя@mail.ru>',
  'phone': '<Телефонный номер пользователя>',
  'first_name': '<Имя пользователя>',
  'last_name': '<Фамилия пользователя>',
  'avatar': '<http://абсолютная/ссылка/на/файл/аватара.ext>',
  'organization_set': <Список имён организаций [
    'Организация 1',
    ...,
    'Организация n'
  ]>
}
```

Получить список пользователей: [/users](#)

Метод: **get**

headers:

```
{  
  'Authorization': 'Bearer <JWT>',  
  'Content-Type': 'application/json'  
}
```

response:

```
{
  'users': [
    {
      'id': <id организации-1>,
      'organizations': <Список организаций [
        {
          'id': <id организации-1>,
          'name': '<Название организации-1>',
          'description': '<Описание организации-1>',
          'users': <Список email-ов пользователей организации-1
            [ 'email пользователя-1', ..., 'email пользователя-n' ]>
        },
        ...
      ]>,
      'name': '<Имя организации-1>',
      'description': '<Описание организации-1>'
    },
    ...,
  ],
  'previous_url': '<Ссылка на предыдущую страницу>',
  'next_url': '<Ссылка на следующую страницу>',
  'page_links': <Список ссылок на страницы [
    [ '<url>', <Номер страницы>, <Активная?: bool>, <Рабочая?: bool> ],
    ...,
  ]>
}
```