

Единая авторизация ТПУ

<https://oauth.tpu.ru>

Данный сервис позволяет провести авторизацию пользователей ТПУ в приложении, сохранив при этом учетные данные пользователей.

1. Настройка приложения

1. Создать приложение (перейти по ссылке <https://api.tpu.ru/dashboard>).
2. Создать платформу и указать версию приложения (получить **api_key**).
3. Настроить авторизацию через OAuth (получить **client_id** и **client_secret**) - указать доверительные домены (каждый на отдельной строке)
4. Дождаться проверки приложения со стороны ТПУ и начать пользоваться.

2. Авторизация

Необходимо перенаправить браузер пользователя по адресу <https://oauth.tpu.ru/authorize>, передав следующие параметры:

client_id обязательный	Идентификатор Вашего приложения.
redirect_uri обязательный	Адрес, на который будет передан code (домен указанного адреса должен соответствовать основному домену в настройках приложения и перечисленным значениям в списке доверенных redirect uri — адреса сравниваются до корневого домена).
response_type	Тип ответа, который Вы хотите получить. Укажите code .
state	Произвольная строка, которая будет возвращена вместе с результатом авторизации. для проверки что отправил, и что вернулось

Пример запроса

```
https://oauth.tpu.ru/authorize?client_id=1&redirect_uri=http://example.com/callback&response_type=code&state=bdc1c79ecb83c00122d24a77e06aa5dc16c8280f7541e89a32108659c353f5
```

3. Ввод корпоративных данных

Если пользователь уже авторизован, то браузер будет перенаправлен по адресу **redirect_uri**, в противном случае пользователю будет предложено ввести свой корпоративный логин и пароль в системе.

4. Получение code

После успешной авторизации приложения браузер пользователя будет перенаправлен по адресу **redirect_uri**, указанному при открытии диалога авторизации. При этом код для получения ключа доступа **code** будет передан как GET-параметр:

```
REDIRECT_URI?code=7a6fa4dff77a228eeda56603b8f53806c883f011c40b72630bb50df056f6479
```

Параметр **code** может быть использован в течение 1 часа для получения ключа доступа к API **access_token** с Вашего сервера.

В случае возникновения ошибки браузер пользователя будет перенаправлен с кодом и описанием ошибки:

```
REDIRECT_URI?error=invalid_request&error_description=error_text
```

5. Получение access_token

Для получения **access_token** необходимо выполнить запрос с Вашего сервера на https://oauth.tpu.ru/access_token, передав следующие параметры **методом POST**:

client_id обязательный	Идентификатор Вашего приложения
client_secret обязательный	Защищенный ключ Вашего приложения (указан в настройках приложения)
redirect_uri НЕ обязательный	URL, который использовался при получении code на первом этапе авторизации. Должен быть аналогичен переданному при авторизации.
code обязательный	Временный код, полученный после прохождения авторизации.

grant_type обязательный	Укажите значение authorization_code
-----------------------------------	--

Пример запроса:

```
https://oauth.tpu.ru/access_token?client_id=1&client_secret=H2PkHm&redirect_uri=http://mysite.ru&code=7a6fa4dff77a228eeda56603b8f53806c883f011c40b72630bb50df056f6479&grant_type=authorization_code
```

В результате выполнения данного запроса Ваш сервер получит вновь созданный **access_token**. Вместе с **access_token** серверу возвращается время жизни ключа **expires_in** в секундах. **expires_in** по умолчанию 24 часа. Процедуру авторизации приложения необходимо повторять в случае истечения срока действия **access_token**, смены пользователем своего логина или пароля или удаления приложения из настроек. Также в ответе присутствует **refresh_token** - срок годности 1 неделя, с помощью его можно получить новый **access_token** не проводя процедуру авторизации повторно.

```
{"access_token":"533bacf01e11f55b536a565b57531ac114461ae8736d6506a3",
refresh_token: "5bac5ac114461ae8736d6506a3b536a565b57531f01e11f533",
"expires_in":43200, "user_id":163098}
```

В случае ошибки будут переданы параметры **error** и **error_description**.

```
{"error":"invalid_grant","error_description":"Code is expired."}
```

6. Получение данных авторизованного пользователя

Получение данных о пользователе необходимо выполнить запрос по адресу <https://api.tpu.ru/v2/auth/user>. передав при этом **методом GET** следующие данные :

apiKey обязательный	Публичный ключ Вашего приложения, соответствующей платформы
access_token обязательный	Ключ доступа авторизованного пользователя в текущей сессии.

В случае успеха вам придёт ответ в виде json структуры, содержащий данные авторизованного пользователя.

```
{"user_id": 163098, "email": "user@tpu.ru", "lichnost": {"familiya": "Иванов", "imya": "Иван"}}
```

7. Выход (завершение сессии пользователя)

Для успешной завершении сессии пользователя нужно сделать редирект на https://oauth.tpu.ru/auth/logout?redirect=<redirect_url>

redirect	URL для обратного редиректа
-----------------	-----------------------------

Вместо заключения

В вашем приложении вы должны хранить в сессии идентификатор авторизованного пользователя и **access_token**. Для получения других данных о пользователе вы можете воспользоваться доступными методами RESTful API.

Пример реализации клиента авторизации на PHP Yii2

```
class TPU extends \yii\authclient\OAuth2
{
    /**
     * {@inheritdoc}
     */
    public $authUrl = 'https://oauth.tpu.ru/authorize';
    /**
     * {@inheritdoc}
     */
    public $tokenUrl = 'https://oauth.tpu.ru/access_token';

    /**
     * @var string
     */
    public $api_key;

    /**
     * {@inheritdoc}
     */
    public $apiBaseUrl = 'https://api.tpu.ru/v2';

    /**
     * @inheritdoc
     */
    protected function initUserAttributes()
    {
        $params = [];
        $params['access_token'] = $this->accessToken->getToken();
        $params['apiKey'] = $this->api_key;

        return $this->api('auth/user', 'GET', $params);
    }

    /**
     * @inheritdoc
     */
    protected function defaultNormalizeUserAttributeMap()
    {
        return [
            'id' => 'user_id'
        ];
    }
}
```