

## Авторизация на сайте через Google OAuth API

*Круглик Роман Игоревич*

*Приамурский государственный университет им. Шолом-Алейхема*

*Студент*

### Аннотация

В статье рассматривается способ интеграции авторизации на сайте через Google OAuth API. Подробно разобрана регистрация проекта на Google OAuth.

**Ключевые слова:** авторизация, Google OAuth, API, PHP.

## Authorization on the site through the Google OAuth API

*Kruglik Roman Igorevich*

*Sholom-Aleichem Priamursky State University*

*Student*

### Abstract

In article discusses how to integrate authorization on a site through the Google OAuth API. Details of the registration of the project on Google OAuth.

**Keywords:** authorization, Google OAuth, API, PHP.

Интернет-пользователь часто сталкивается с проблемой хранения паролей и логинов от различных систем, где он регистрируется. Каждый второй сайт требует регистрации, а неоднократное использование одного и того же пароля не безопасно и может привести к потере доступа.

Для решения этой проблемы можно использовать API авторизацию Google OAuth. С помощью этого инструмента пользователь может авторизоваться на веб-ресурсе с ранее созданным аккаунтом Google.

Исследования в области авторизации через Google API актуальны и по сей день. В статье [1] А.С. Мосягина рассматривается принцип работы авторизации в сервисах компании Google на сторонних ресурсах. Так же И.С. Красильников [2] рассматривается функции для авторизации пользователя в сервисах компании Google на сторонних сайтах. В статье А.С. Мосягин [3] поэтапно расписана разработка модуля авторизации в сервисах google на языке программирования Java

В данной статье будет рассмотрен способ авторизации пользователей на сайте с помощью Google OAuth API.

Для начала необходимо зарегистрировать проект в Google API. Для этого необходимо зайти в консоль разработчиков и перейти в панель управления Google (см. рис. 1).

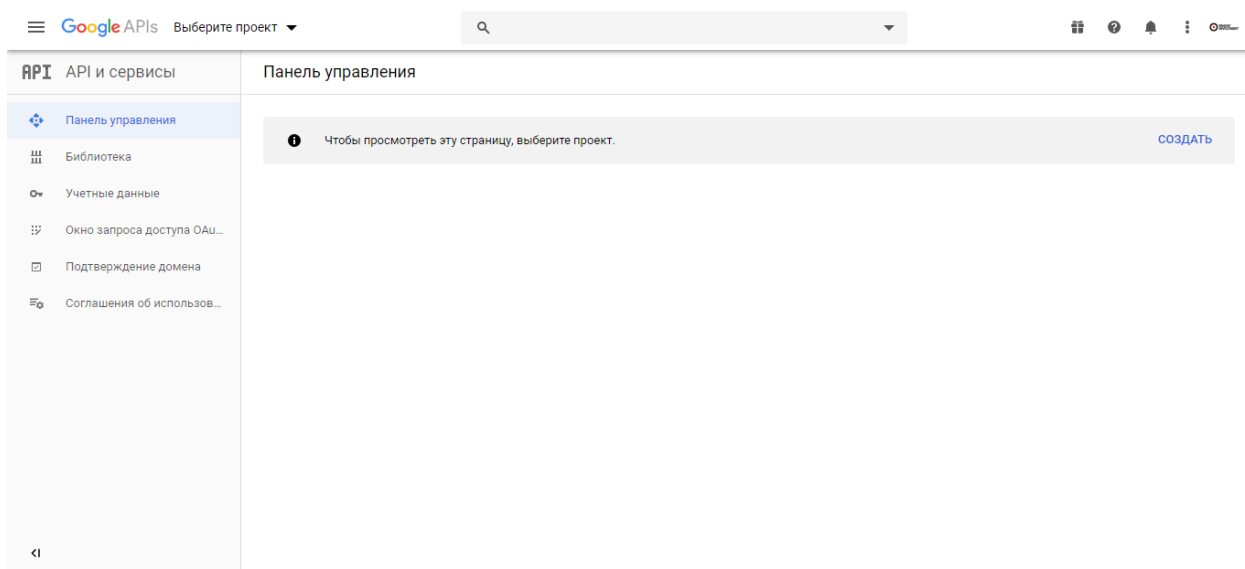


Рисунок 1. Панель управления

Нажимаем на «Выберите проект» и откроется всплывающее окно (см. рис. 2).

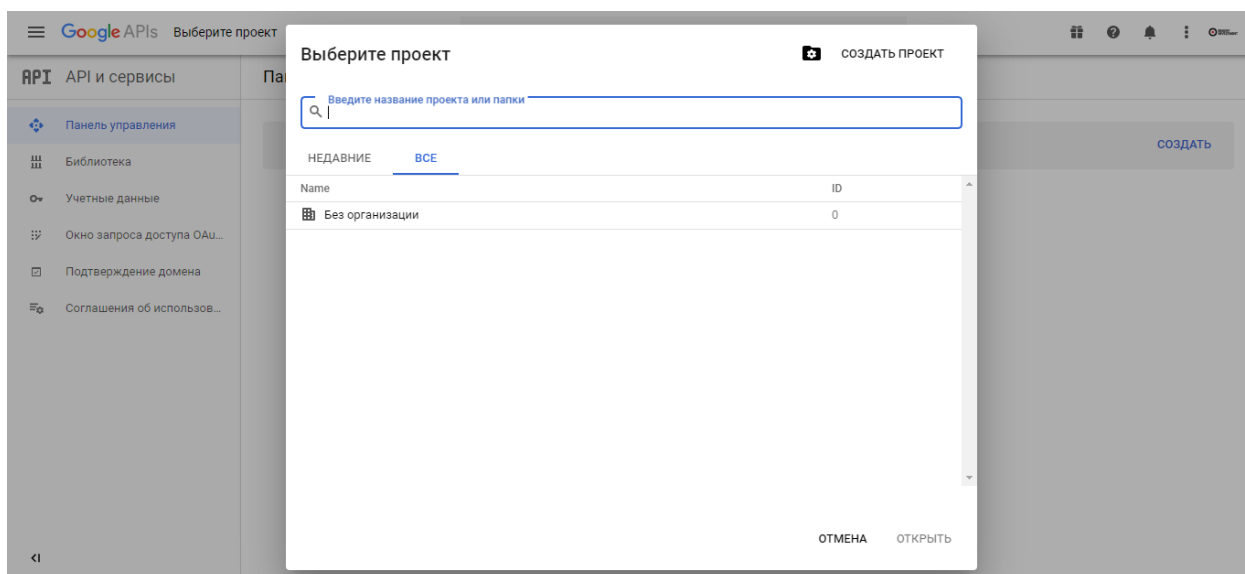


Рисунок 2. Окно для выбора проекта

Нажимаем на «Создать проект» и заполняем название проекта (см. рис. 3).

Меню навигации  
Создание проекта

⚠ Доступный остаток квоты на projects: 12. Отправьте запрос на увеличение квоты или удалите проекты. [Подробнее...](#)

[MANAGE QUOTAS](#)

Название проекта \*  
Test

Идентификатор проекта: inductive-world-265711. Его нельзя будет изменить позже.

[ИЗМЕНИТЬ](#)

Местоположение \*  
Без организации [ОБЗОР](#)

Родительская организация или папка

[СОЗДАТЬ](#) [ОТМЕНА](#)

Рисунок 3. Создание проекта

Нажимаем на «Создать проект» и тем самым проект будет сохранён и вас перенаправит на главную страницу. Теперь необходимо перейти на вкладку «Окно запроса доступа OAuth», которая находится в левой панели (см. рис. 4).

API API и сервисы

Панель управления  
Библиотека  
Учётные данные  
Окно запроса доступа OAuth...  
Подтверждение домена  
Соглашения об использо...

Окно запроса доступа OAuth

Это окно будет отображаться перед тем, как пользователи пройдут аутентификацию. В нем вы можете запросить у них разрешение на доступ к их личным данным и предоставить ссылки на условия использования и политику конфиденциальности. Заданные здесь настройки действуют для всех приложений в этом проекте.

Состояние подтверждения  
Не опубликовано

Название приложения  
Название приложения, которому требуется доступ.  
Test

Логотип приложения  
Картинка на экране запроса доступа, которая поможет пользователям идентифицировать приложение.  
Локальный файл для загрузки [Обзор](#)

Адрес электронной почты службы поддержки  
Отображается в окне запроса доступа в качестве контактного адреса по вопросам поддержки пользователей.  
olimpictraid@gmail.com

Области действия для API Google

Окно запроса доступа  
В окне запроса доступа пользователи могут узнать, кто и зачем запрашивает доступ к их данным.

Проверка OAuth  
В ряде случаев может потребоваться проверка окна запроса доступа и приложения. Это обязательно для общедоступных приложений, если выполняется хотя бы одно из следующих условий:

- Для приложения выбрана область действия, позволяющая получить доступ к конфиденциальным данным, и (или) область действия с ограниченным доступом.
- В окне запроса доступа OAuth в приложении отображается значок.
- С приложением связано большое количество авторизованных доменов.
- Вы внесли изменения в проверенное окно запроса доступа OAuth.

Проверка может занять несколько недель. Мы будем информировать вас о ходе проверки по электронной почте. [Подробнее о проверке...](#)

Пока окно запроса доступа и приложение не

Рисунок 4. Окно запроса доступа OAuth

Необходимо заполнить поля и нажать кнопку «Сохранить». Далее перейти на вкладку «Учётные данные» и нажать кнопку «Создать учётные данные», и в выпадающем меню выбрать «Идентификатор клиента OAuth» (см. рис. 5).

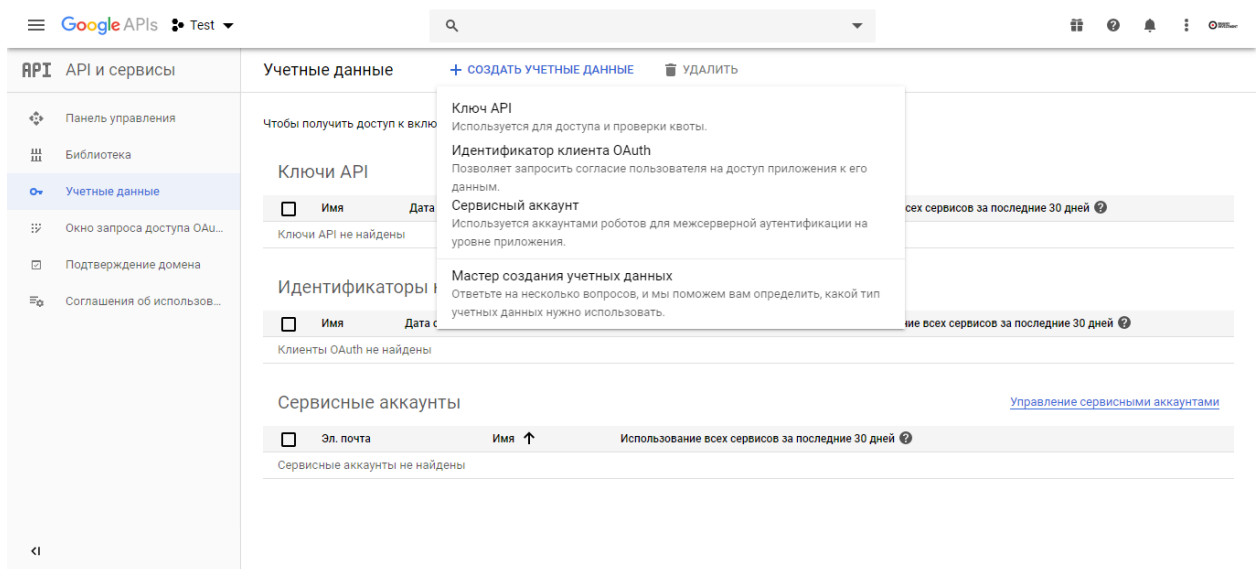


Рисунок 5. Настройка учётных данных

Далее выбираем «Веб-приложение» и указывает адрес на котором будет происходить авторизация (см. рис. 6).

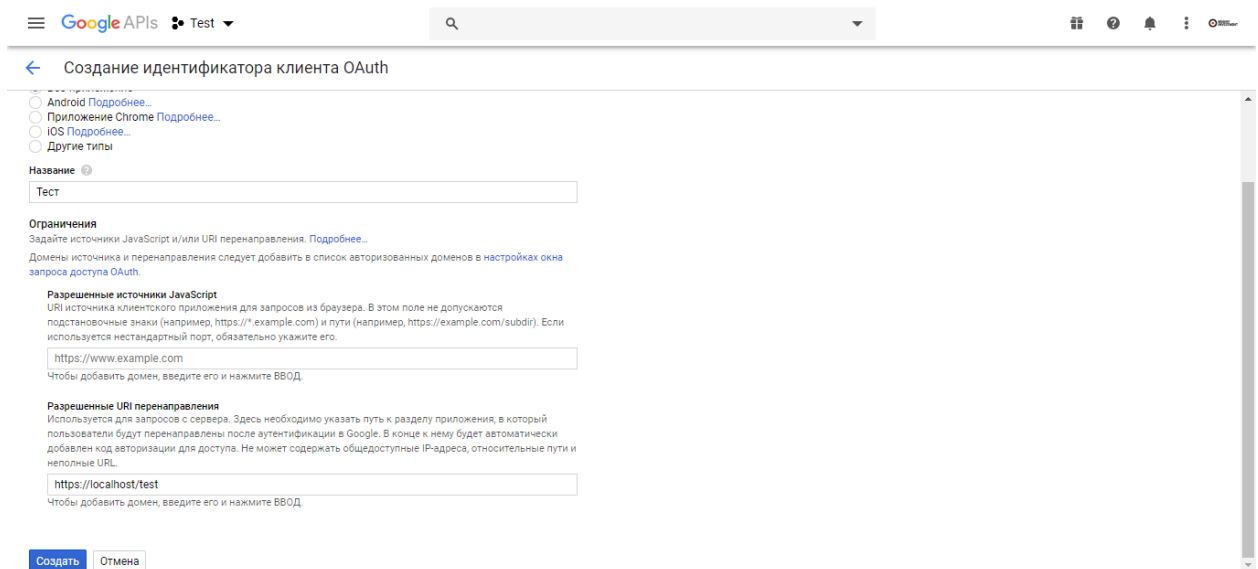


Рисунок 6. Создание OAuth

Настройка проекта закончена, теперь необходимо скачать клиентскую библиотеку Google PHP API с помощью composer. Вводим в консоль приведённый код:

```
$composer require google/apiclient:"^2.0"
```

После успешного скачивания рассмотрим файл index.php, где происходит подключение и работа с библиотекой (см. рис. 7).

```
<?php
require_once 'vendor/autoload.php';
$clientID = '<YOUR_CLIENT_ID>';
$clientSecret = '<YOUR_CLIENT_SECRET>';
$redirectUri = '<REDIRECT_URI>';
$client = new Google_Client();
$client->setClientId($clientID);
$client->setClientSecret($clientSecret);
$client->setRedirectUri($redirectUri);
$client->addScope( scope_or_scopes: "email");
$client->addScope( scope_or_scopes: "profile");
if (isset($_GET['code'])) {
    $token = $client->fetchAccessTokenWithAuthCode($_GET['code']);
    $client->setAccessToken($token['access_token']);
    $google_oauth = new Google_Service_Oauth2($client);
    $google_account_info = $google_oauth->userinfo->get();
    $email = $google_account_info->email;
    $name = $google_account_info->name;
} else {
    echo "<a href='". $client->createAuthUrl(). "'>Google Login</a>";
}
?>
```

Рисунок 7. Файл index.php

В файле необходимо изменить 3 параметра, с помощью которых происходит интеграция с сервисами Google.

1. clientId - Идентификатор клиента,
2. clientSecret - Секретный код клиента,
3. redirectUri – URL на котором будет происходить авторизация.

Все эти данные берутся из консоли разработчика, где мы авторизовали проект. Посмотрим на результат интеграции авторизации. Если ошибок нет, то всё сделано правильно (см. рис. 6).

### [Google Login](#)

Рисунок 6. Результат

При нажатии на ссылку «Google Login» пользователя перенаправит на страницу авторизации в Google сервис. Если авторизация прошла успешно, система получит доступ к информации пользователя и может использовать её при регистрации нового аккаунта.

В результате новые пользователи системы могут, не проходя форму регистрации быстро авторизоваться на сайте и пользоваться функционалом.

**Библиографический список**

1. Мосягин А.С. Рассмотрение принципов авторизации в системе Google с примерами // Аллея науки. 2018. Т. 3. № 7 (23). С. 119-124.
2. Красильников И.С. Разработка функций для авторизации пользователя в сервисах компании google // XX Всероссийская студенческая научно-практическая конференция Нижневартковского государственного университета сборник статей. Ответственный редактор А.В. Коричко. 2018. С. 188-190.
3. Мосягин А.С. разработка модуля авторизации в сервисах google на языке программирования java // Молодой исследователь: вызовы и перспективы Сборник статей по материалам LXXX международной научно-практической конференции. 2018. С. 123-127.