## ****Google OAuth 2.0: Adımlar****

#### **1. **Google Project Console’dan gerekli ayarları yap:****

* [https://console.developers.google.com/](https://console.developers.google.com/" \t "_blank) adresine gir.
* Yeni bir “OAuth client ID” oluştur.
* “Authorized redirect URIs” kısmına kendi backend’inde belirleyeceğin /auth/google/callback adresini ekle (örneğin: http://localhost:3000/auth/google/callback).

#### **2. **Client ID ve Client Secret değerlerini al.****

#### **3. **Backend’de Google Auth rotasını kur:****

1. Kullanıcıyı Google’a yönlendir.
2. Google’dan dönen code’u alıp, access\_token’a çevir.
3. access\_token ile kullanıcı bilgisini çek.
4. Buradan kendi sisteminde oturum aç.

## ****Express/Node.js ÖRNEĞİ:****

### ****.env dosyası:****

bash

GOOGLE\_CLIENT\_ID=... (Google Console'dan)

GOOGLE\_CLIENT\_SECRET=... (Google Console'dan)

GOOGLE\_REDIRECT\_URI=http://localhost:3000/auth/google/callback

SESSION\_SECRET=çok\_gizli

### ****app.js (Express ile En Temel OAuth Akışı):****

javascript

const express = require('express');const session = require('express-session');const axios = require('axios');require('dotenv').config();

const app = express();

app.use(session({

secret: process.env.SESSION\_SECRET,

resave: false,

saveUninitialized: true

}));

// 1. Google'a yönlendiren endpoint

app.get('/auth/google', (req, res) => {

const scope = [

'profile',

'email'

].join(' ');

const redirect = `https://accounts.google.com/o/oauth2/v2/auth?` +

`client\_id=${process.env.GOOGLE\_CLIENT\_ID}` +

`&redirect\_uri=${process.env.GOOGLE\_REDIRECT\_URI}` +

`&response\_type=code` +

`&scope=${scope}` +

`&access\_type=offline`;

res.redirect(redirect);

});

// 2. Callback endpoint, burada Google'ın gönderdiği code ile token alıp bilgi çekiyoruz

app.get('/auth/google/callback', async (req, res) => {

const code = req.query.code;

if (!code) return res.status(400).send('No code!');

// Exchange code for tokens

try {

const tokenRes = await axios.post('https://oauth2.googleapis.com/token', null, {

params: {

code,

client\_id: process.env.GOOGLE\_CLIENT\_ID,

client\_secret: process.env.GOOGLE\_CLIENT\_SECRET,

redirect\_uri: process.env.GOOGLE\_REDIRECT\_URI,

grant\_type: 'authorization\_code'

},

headers: {

'Content-Type': 'application/x-www-form-urlencoded',

},

});

const { access\_token, id\_token } = tokenRes.data;

// Access Token ile User Info alalım

const userInfo = await axios.get('https://www.googleapis.com/oauth2/v2/userinfo', {

headers: {

Authorization: `Bearer ${access\_token}`

}

});

// Kullanıcı bilgisini session'a koyup anasayfaya yönlendir (ya da kendi user auth sisteminle eşleştir)

req.session.user = userInfo.data; // burada email, isim, foto vs gelir.

res.redirect('/profile');

} catch (err) {

console.error(err.response?.data || err.message);

res.status(500).send('Google Auth Error.');

}

});

// 3. Kullanıcı profili (oturum açmış ise)

app.get('/profile', (req, res) => {

if (!req.session.user) return res.redirect('/auth/google');

res.send(`

<h2>Merhaba, ${req.session.user.name}</h2>

<img src="${req.session.user.picture}" width="80"/>

<p>Email: ${req.session.user.email}</p>

<a href="/logout">Çıkış Yap</a>

`);

});

// 4. Logout

app.get('/logout', (req, res) => {

req.session.destroy();

res.redirect('/');

});

app.listen(3000, () => console.log("http://localhost:3000 hazır"));

### ****Basit Kullanım****

Tarayıcıda  
http://localhost:3000/auth/google  
adresine giden kullanıcı, Google oturum açma ekranına yönlenir.  
İzin verdiğinde senin /auth/google/callback endpointine gelir ve login olur.

### ****FRONTEND (Buton için):****

html

<a href="/auth/google">

<button>

<svg ...> <!-- Google logo ikonun burada olabilir --></svg>

Google ile Giriş Yap

</button></a>

## ****Kısa Notlar****

* VSCode ile bu örneği, .env ve app.js ile çalıştırabilirsin.
* Bu örnek tamamen vanilla backend ile, hiçbir ek auth kütüphanesi olmadan çalışır.
* Kendi veritabanına (kullanıcı datasına) bağlamak istersen, userInfo.data ile user’ı eşleştirip ID verebilirsin.
* “redirect\_uri” çok önemli; Google Console’daki ve backend kodundaki URI’ların birebir eşleşmesi gerekir.