

VYTAUTAS MAGNUS UNIVERSITY
FACULTY OF INFORMATICS
DEPARTMENT OF APPLIED INFORMATICS

INTERNET TECHNOLOGY (INF3002_EN)

Google API Integration

Student:Dmytro Vodianytskyi IF2100075

Lab assistant: Vytautas Barzdaitis

Kaunas, 2023

Contents

| | | |
|----------|--|----------|
| 1 | Introduction | 3 |
| 2 | Prerequisites | 3 |
| 3 | Integration Steps | 3 |
| 3.1 | Create a Project in Google Cloud Console | 3 |
| 3.2 | Enable Google API | 3 |
| 3.3 | Create OAuth 2.0 Client ID | 3 |
| 3.4 | Install Required Packages | 4 |
| 4 | Finall seps | 4 |
| 5 | How it works on polls site | 4 |
| 5.1 | Login with Google | 4 |
| 5.2 | Accessing User Information | 4 |
| 5.3 | Logout | 4 |
| 5.4 | Error Handling | 4 |

1 Introduction

This documentation provides information on how to integrate Google API into website. The integration will allow users to sign in to your website using their Google account.

2 Prerequisites

Before proceeding with the integration, you need to have the following:

- A Google account
- A project in the Google Cloud Console
- Access to your website's codebase

3 Integration Steps

Follow these steps to integrate Google API into your website:

3.1 Create a Project in Google Cloud Console

1. Log in to your Google account and go to the Google Cloud Console <https://console.cloud.google.com/>.
2. Click on the "Select a Project" dropdown menu at the top of the screen and click on "New Project".
3. Enter a name for your project and click on the "Create" button.
4. Once the project is created, select it from the dropdown menu at the top of the screen.

3.2 Enable Google API

1. In the Google Cloud Console, go to the "APIs Services" section and click on "Dashboard".
2. Click on the "+ ENABLE APIS AND SERVICES" button at the top of the screen.
3. Search for "Google+ API" and select it from the list of results.
4. Click on the "Enable" button to enable the API.

3.3 Create OAuth 2.0 Client ID

1. In the Google Cloud Console, go to the "APIs Services" section and click on "Credentials".
2. Click on the "+ CREATE CREDENTIALS" button and select "OAuth client ID".
3. Select "Web application" as the application type.
4. Enter a name for your client ID.
5. Enter the authorized JavaScript origins for your website. This is the URL where your website is hosted.
6. Enter the authorized redirect URIs. This is the URL where users will be redirected after signing in with Google.
7. Click on the "Create" button to create the client ID.
8. Copy the client ID and client secret. You will need these later in the integration process.

3.4 Install Required Packages

You need to install the following packages to integrate Google API into your website:

- **social-auth-app-django**: This package provides integration with social authentication providers, including Google.
- **social-auth-core**: This package provides the core functionality for social authentication.

You can install these packages using pip:

```
pip install social-auth-app-django social-auth-core
```

These settings will allow your Django application to authenticate with the Google OAuth2 API and retrieve user information.

4 Finall seps

Finally, you need to make sure that your Django application is set up to handle the OAuth2 callback correctly. This involves creating a URL pattern that matches the authorized redirect URL that you specified in the Google Cloud Console, and configuring the appropriate views to handle the callback. In your case, you've already added the necessary URL pattern and views to your Django application, so you should be all set!

5 How it works on polls site

A couple of points to understand how it works

5.1 Login with Google

When a user clicks on the "Login with Google" button, it redirects the user to the Google login page where they enter their Google credentials. After successful login, the user is redirected back to your site with a token that confirms their identity.

5.2 Accessing User Information

With the user's identity token, your site can access the user's Google profile information, such as their name and email address. This information can be used to pre-fill forms, create user accounts, or personalize the user's experience on your site.

5.3 Logout

When the user clicks on the "Logout" button, they are logged out of your site and their Google session is terminated.

5.4 Error Handling

If there is an error in the authentication process, such as an incorrect password or an invalid token, an error message is displayed to the user explaining the issue.