

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»**

**Факультет інформаційно-комп'ютерних технологій
Кафедра комп'ютерної інженерії та кібербезпеки**

ЗВІТ З ПЕРЕДДИПЛОМНОЇ ПРАКТИКИ

студента 4-го курсу групи КБ-19-2
спеціальності 125 «Кібербезпека»
Москвіна Дмитра Володимировича

Керівник: Окунькова Оксана Олексіївна
старший викладач кафедри
комп'ютерної інженерії та кібербезпеки

Дата захисту: “_____” _____ 2023 р.
Національна шкала: _____
Кількість балів: _____
Оцінка ECTS: _____

Члени комісії:

_____ О.О. Окунькова
_____ Воротніков В.В.
_____ Россінський Ю.М.

м. Житомир – 2023 рік

РЕФЕРАТ

Звіт з переддипломної практики складається зі вступу, двох розділів, висновків, списку умовних скорочень, списку використаної літератури.

Текстова частина викладена на 21 сторінках друкованого тексту.

Список використаних джерел містить 10 найменувань і займає 1 сторінку. В роботі наведено 25 рисунків.

В першому розділі було коротко розглянуто місце проходження практики, історію та призначення компанії.

В другому розділі було проведено аналіз роботи, ознайомлено з теоретичними відомостями та виконано практичні завдання.

Висновки містять в собі результати виконаної роботи.

Ключові слова: ЕЛЕКТРОННА КОМЕРЦІЯ, AUTHENTICATION, AUTHENTICATION PROVIDER, SINGLE SIGN-ON, GOOGLE IDENTITY, OAUTH 2.0, OPENID CONNECT, JWT TOKEN.

					«Житомирська політехніка».23.125.09.000 – ПР						
Змн.	Арк.	№ докум.	Підпис	Дата	Звіт з переддипломної практики			Літ.	Арк.	Аркушів	
Розроб.		Москвін Д. В.								4	28
Керівник		Окунькова О.О.						ФІКТ, гр. КБ-19-2			
Реценз.											
Н. Контр.											
Затверд.		Єфіменко А.А.									

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	4
ВСТУП.....	5
РОЗДІЛ 1. ВІДОМОСТІ ПРО МІСЦЕ ПРОХОДЖЕННЯ ПРАКТИКИ.....	6
1.1. Sana Commerce.....	6
1.2. Історія створення компанії.	7
1.3. Поняття електронної комерції.....	7
1.4. Sana Commerce Cloud.	8
РОЗДІЛ 2. ХАРАКТЕРИСТИКА ТА ПЕРЕЛІК ВИКОНАНИХ РОБІТ	10
2.1. Практичні завдання на практиці.....	10
ВИСНОВКИ	27
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	28

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПЗ	Арк.
		Окунькова О.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		3

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

URI – Уніфікований ідентифікатор ресурсів

API – Інтерфейс прикладного програмування

SSO – Технологія єдиного входу (Single Sign-On)

JWT – JSON Web Token

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				4
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Переддипломна практика допомагає визначитися з майбутньою професією, закріпити уявлення про роботу у обраній сфері та отримати базові практичні навички роботи з обладнанням, програмним інструментарієм та комунікації з колегами.

Об'єктом проходження практики є відділ розробки програмного забезпечення товариства з обмеженою відповідальністю «Сана Комерс Україна».

Предметом практики є набуття практичних навичок у роботі з теоретичним матеріалом та його застосування на практиці, отримання досвіду роботи з великим проектом та комунікації з колегами.

Основними завданнями проходження виробничої практики в компанії «Сана Комерс Україна» є:

- ознайомлення з програмними продуктами компанії, її напрямками;
- отримання практичних навичок роботи у команді над великим проектом, розробка програмного забезпечення, комунікація з колегами.

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1. ВІДОМОСТІ ПРО МІСЦЕ ПРОХОДЖЕННЯ ПРАКТИКИ

1.1. Sana Commerce.

Sana Commerce — це компанія, що займається електронною комерцією B2B та програмним забезпеченням для електронної комерції B2C зі штаб-квартирою на заводі Van Nelle, розташованому в Роттердамі, Нідерланди. Sana Commerce була заснована в 2008 році і є дочірньою компанією ISM eCompany, яку заснували Ян Кіс де Ягер і Карел ван дер Вуде в 1992 році.



Рисунок 1.1 – Логотип Sana Commerce



Рисунок 1.2 – Завод Van Nelle, штаб-квартира Sana Commerce

Будучи Золотим сертифікованим партнером Майкрософт (Microsoft Gold Certified Partner), Sana Commerce є незалежним розробником програмного забезпечення (ISV), який займається пошуком вдалих рішень для збільшення прибутковості та продуктивності веб-шопів клієнтів. Sana Commerce має офіси продажів та технічної підтримки в Нідерландах, а також власний офіс, що займається безпосередньо розробкою в Україні. Sana Commerce пропонує рішення для електронної комерції, що повністю інтегруються у Microsoft Dynamics та SAP ERP, включаючи SAP Business Suite.

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				6
Змн.	Арк.	№ докум.	Підпис	Дата		

1.2. Історія створення компанії.

До заснування Sana Commerce компанія ISM eCompany використовувала SANA (Мережева архітектура програмного забезпечення) як програмну платформу для веб-додатків і веб-сайтів для керування вмістом. У 2008 році це програмне забезпечення було розроблено в інтегроване рішення для електронної комерції та отримало назву Sana Software. З самого початку Sana Software була доступна для Microsoft Dynamics NAV і Microsoft Dynamics AX. У 2013 році назва бренду була змінена на Sana Commerce. З 2014 року Sana Commerce також доступна для SAP ERP і названа партнером SAP.

1.3. Поняття електронної комерції.

Електронна комерція— це сфера цифрової економіки, що включає всі фінансові та торгові транзакції, які проводяться за допомогою комп'ютерних мереж, та бізнес-процеси, пов'язані з проведенням цих транзакцій.

Основними складовими електронної комерції є мобільна комерція, електронний переказ коштів, управління ланцюгами поставок, Інтернет-маркетинг, обробка онлайн-транзакцій, електронний обмін даними (EDI), системи управління запасами та автоматизовані системи збору даних.

Активний розвиток цього напрямку зумовлений технологічними досягненнями напівпровідникової промисловості (електронної промисловості).

У багатьох випадках електронна комерція дозволяє скоротити шлях перепродажу продукту від виробника до споживача. Це можливо завдяки використанню Інтернет-технологій, що надають можливість ефективної прямої взаємодії з кінцевим споживачем, тому компанії можуть виконувати роль, яку традиційно виконували проміжні постачальники. Це також дозволяє накопичувати інформацію про усі продажі та про усіх клієнтів, що у свою чергу дозволяє виконати досконалий бізнес-аналіз та маркетингові дослідження. Це є великою перевагою у конкурентній боротьбі.

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

Електронне середовище широко використовується для доставки цифрового медіа-контенту (музика, фільми, преса тощо), корисної інформації, освітніх матеріалів, а також компаніями-виробниками програмного забезпечення для його продажу.

Найбільшою перевагою електронної комерції є суттєве зниження витрат на оформлення угоди та її подальше обслуговування. Тому бізнес-процеси, які можуть бути переведені на електронну основу мають потенціал зниження витрат на них, що у свою чергу призводить до зниження собівартості товару чи послуги. Найвідомішим прикладом здійснення електронної комерції є Інтернет-магазин, який являє собою веб-ресурс з каталогом продукції та можливістю замовлення і оплати товарів, які сподобались покупцю. Все більше компаній у світі впроваджують рішення електронної комерції у своєму бізнесі.

1.4. Sana Commerce Cloud.

Sana Commerce Cloud допомагає підприємцям керувати інтернет-магазином, який надає клієнтам надійні та точні дані. Платформа досягає цього, поєднуючи систему Microsoft Dynamics ERP (планування ресурсів підприємства) та функції електронної комерції як єдину структуру. В результаті потрібно підтримувати лише одну базу даних із оновленням даних у режимі реального часу.

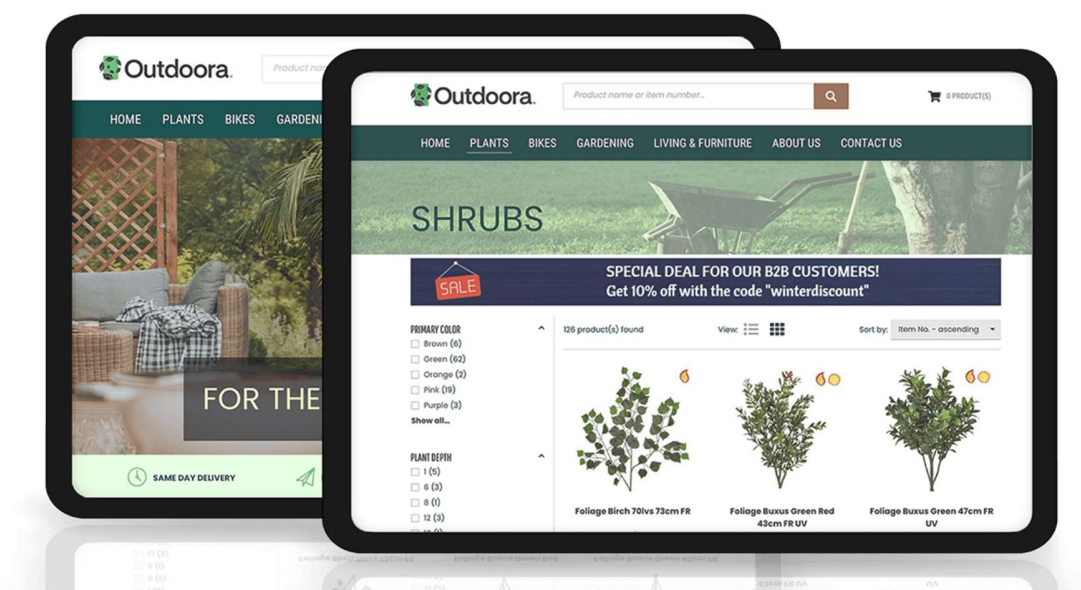


Рисунок 1.3 – Інтернет-магазин на базі Sana Commerce Cloud

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				8
Змн.	Арк.	№ докум.	Підпис	Дата		

Sana Commerce також пропонує:

- гнучкість: Sana Commerce може похвалитися розділеними інтерфейсом та сервером, побудованими на основі React. Це дозволяє далекоглядним компаніям бути готовими до безперешкодної торгівлі та отримувати вигоду з усієї гнучкості цієї структури.
- швидка робота: Sana Commerce Cloud покращує інтерфейс користувача за рахунок високої продуктивності сторінок. Фактично, це перший односторінковий додаток (SPA), спеціально створений для підприємств B2B.

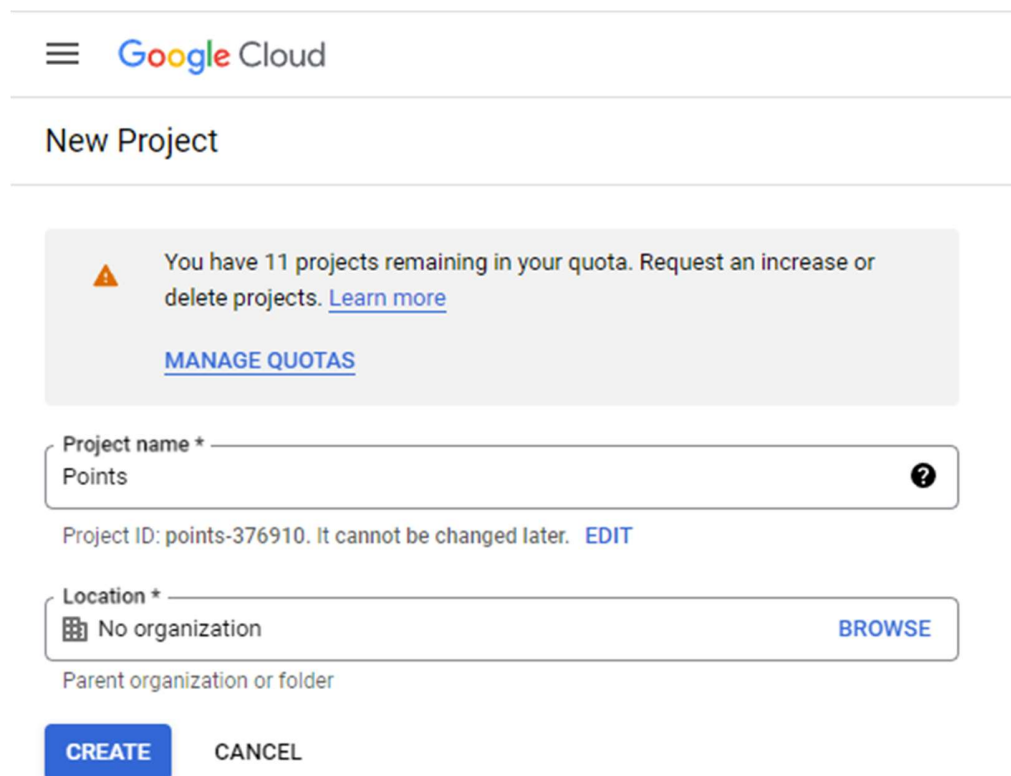
		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2. ХАРАКТЕРИСТИКА ТА ПЕРЕЛІК ВИКОНАНИХ РОБІТ

2.1. Практичні завдання на практиці.

Завдання 1. Налаштувати проект у Google API Console, щоб отримати облікові дані OAuth 2.0, установити URI перенаправлення та налаштувати інформацію на екрані згоди користувача.

Створимо проект у Google API Console:



Google Cloud

New Project

You have 11 projects remaining in your quota. Request an increase or delete projects. [Learn more](#)

[MANAGE QUOTAS](#)

Project name * ?

Project ID: points-376910. It cannot be changed later. [EDIT](#)

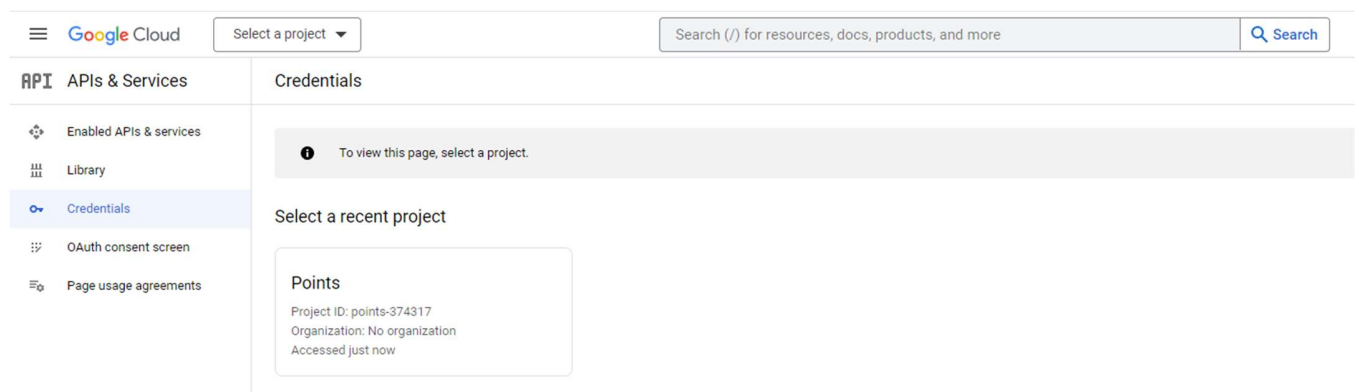
Location * [BROWSE](#)

Parent organization or folder

[CREATE](#) [CANCEL](#)

Рисунок 2.1 – Створення проекту в Google API Console

Перейдемо на сторінку облікових даних:



Google Cloud

Select a project

Search (/) for resources, docs, products, and more

Search

APIs & Services

- Enabled APIs & services
- Library
- Credentials**
- OAuth consent screen
- Page usage agreements

Credentials

To view this page, select a project.

Select a recent project

Points

Project ID: points-374317
Organization: No organization
Accessed just now

Рисунок 2.2 – Сторінка облікових даних

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				10
Змн.	Арк.	№ докум.	Підпис	Дата		

Оберемо створений проєкт та створимо облікові дані OAuth 2.0 client ID:

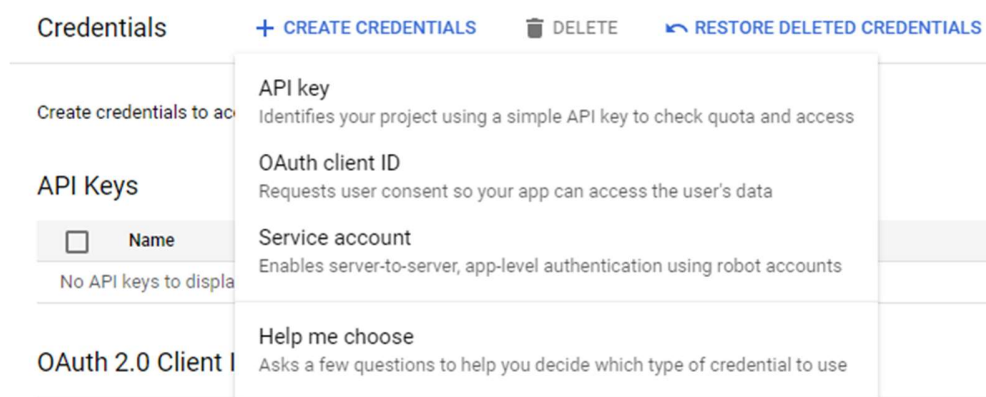


Рисунок 2.3 – Створення облікових даних

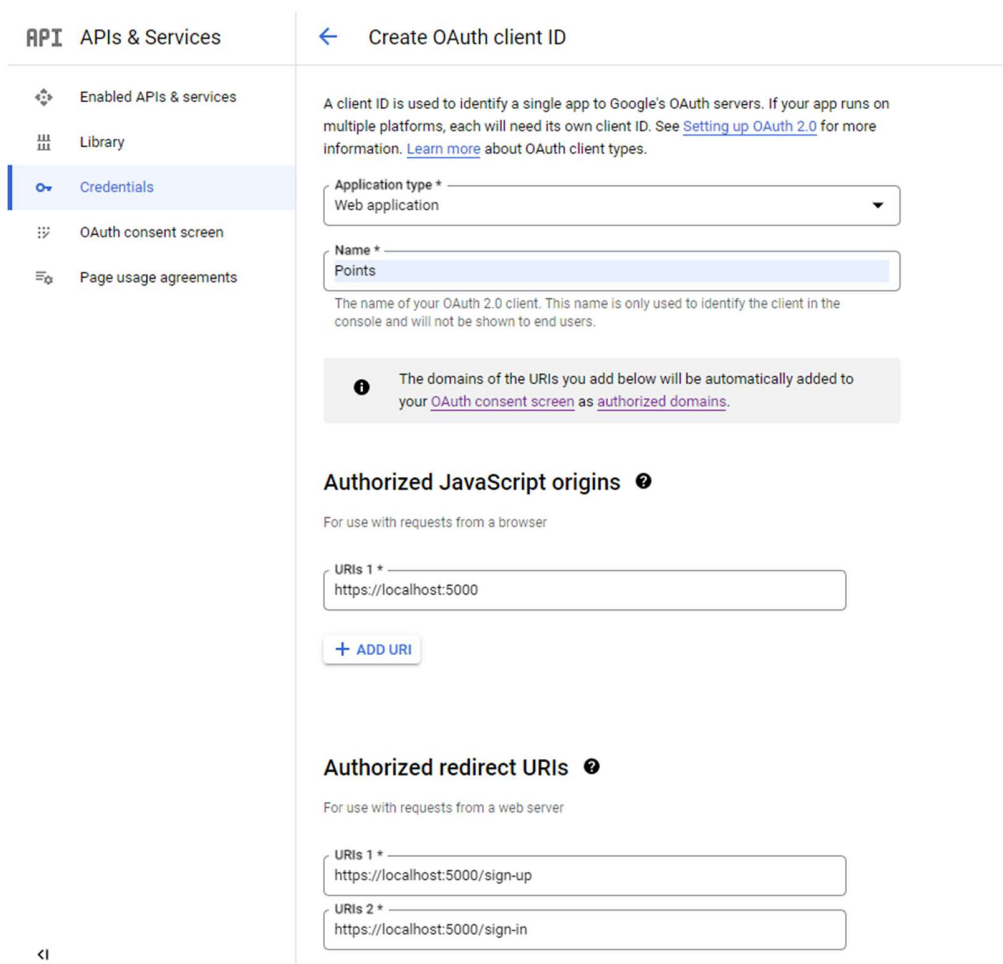


Рисунок 2.4 – Створення облікових даних

Варто зазначити, що ми додали в URI значення localhost:5000, оскільки саме на цьому порті ми будемо локально запускати наш проєкт. У майбутньому можна буде додати інший URI, наприклад на наш задеплойний проєкт у Microsoft Azure.

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				11
Змн.	Арк.	№ докум.	Підпис	Дата		

Налагодимо екран згоди користувача:

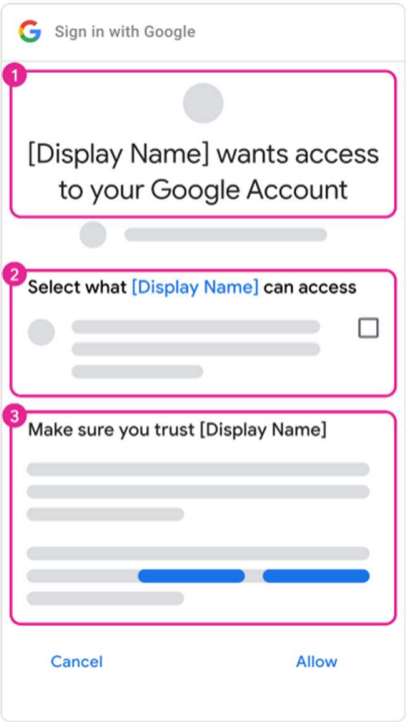


Рисунок 2.5 – Структура екрану згоди користувача.

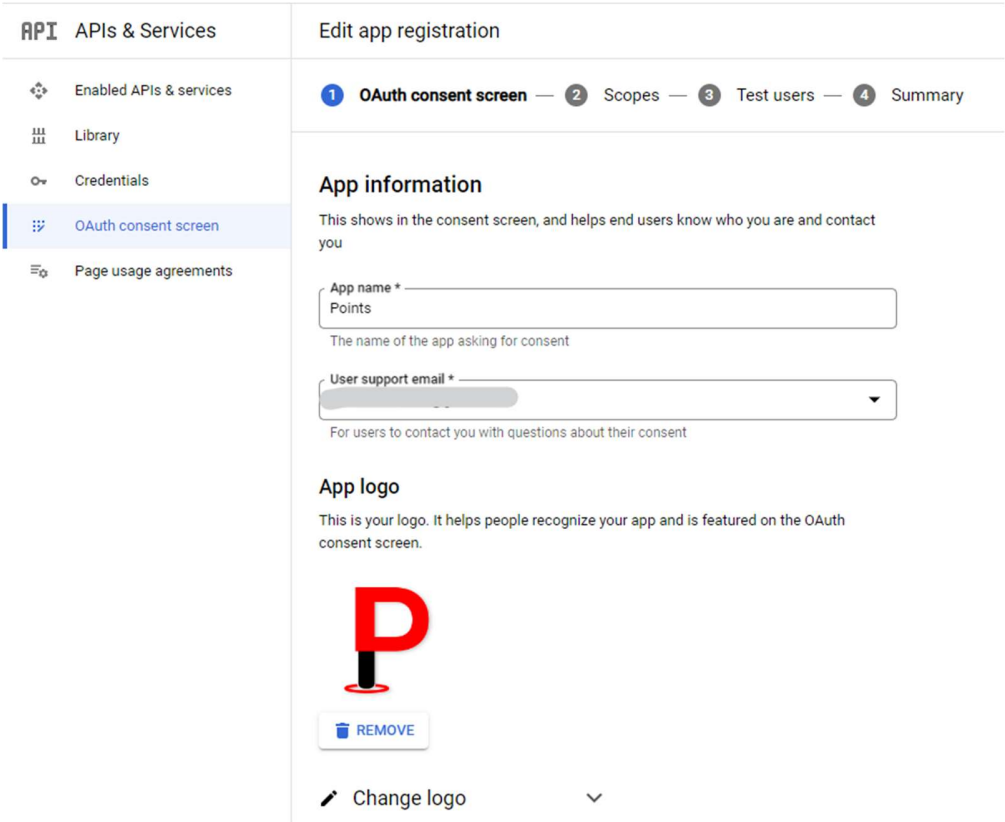


Рисунок 2.6 – Налаштування екрану згоди користувача

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				12
Змн.	Арк.	№ докум.	Підпис	Дата		

API APIs & Services

Enabled APIs & services

Library

Credentials

OAuth consent screen

Page usage agreements

Edit app registration

✓ OAuth consent screen

2 Scopes

3 Test users

4 Summary

Scopes express the permissions you request users to authorize for your app and allow your project to access specific types of private user data from their Google Account. [Learn more](#)

ADD OR REMOVE SCOPES

Your non-sensitive scopes

API ↑	Scope	User-facing description	
	..	See your primary Google Account email address	🗑
	./auth/userinfo	See your personal info, including any personal info you've made publicly available	🗑
	.email	Associate you with your personal info on Google	🗑
	openid		

🔒 Your sensitive scopes

Sensitive scopes are scopes that request access to private user data.

API ↑	Scope	User-facing description
No rows to display		

Рисунок 2.6 – Налаштування екрану згоди користувача

В scopes додамо лише базові пункти, щоб забезпечити можливість доступу до імені, пошти та іншої персональної інформації, яка нам буде необхідна для реалізації функціоналу.

Додамо тестового користувача, адже для повноцінного функціонування провайдеру аутентифікації потрібно опублікувати проект та пройти відповідні перевірки безпеки від Google. Наш проект є навчальним, тому обмежимося тестовим режимом та обмеженим списком дозволених користувачів:

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				13
Змн.	Арк.	№ докум.	Підпис	Дата		

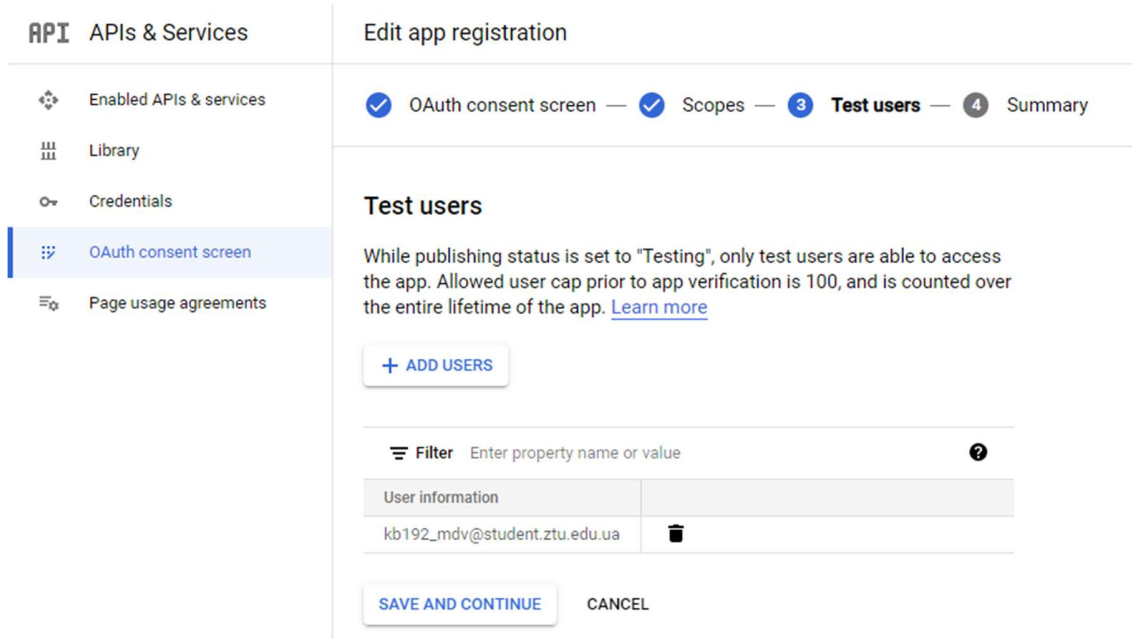


Рисунок 2.7 – Налаштування екрану згоди користувача

Тепер наш проєкт в Google API Console повністю готовий для роботи, наступним кроком буде розробити веб-додаток, запустити його на localhost:5000 та реалізувати сторінки логіну та реєстрації з URL, які відповідають до сконфігурованим значенням.

Завдання 2. Розробити веб-додаток з системою аутентифікації на базі технології єдиного входу Google Identity, який повинен містити функціонал логіну, реєстрації користувача та відкликання доступу до даних профіля в провайдері аутентифікації.

Аутентифікація користувача передбачає отримання ідентифікаційного маркера та його перевірку. Ідентифікаційні маркери — це стандартизована функція OpenID Connect, розроблена для використання в обміні підтвердженнями ідентифікації в Інтернеті.

Найбільш часто використовувані підходи для автентифікації користувача та отримання ідентифікаційного маркера називаються «серверним» потоком і «неявним» потоком. Потік сервера дозволяє внутрішньому серверу програми перевірити особу особи за допомогою браузера або мобільного пристрою. Неявний потік використовується, коли програмі на стороні клієнта (як правило, програмі JavaScript, що працює в браузері) потрібно отримати доступ до API безпосередньо, а не через внутрішній сервер.

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				14
Змн.	Арк.	№ докум.	Підпис	Дата		

Ми реалізуємо серверний потік для автентифікації користувача, оскільки неявний потік значно складніший через ризики безпеки під час обробки та використання токенів на стороні клієнта.

Створимо проект з сервером .NET 6.0 та клієнтом на React.js за допомогою відповідного шаблону:

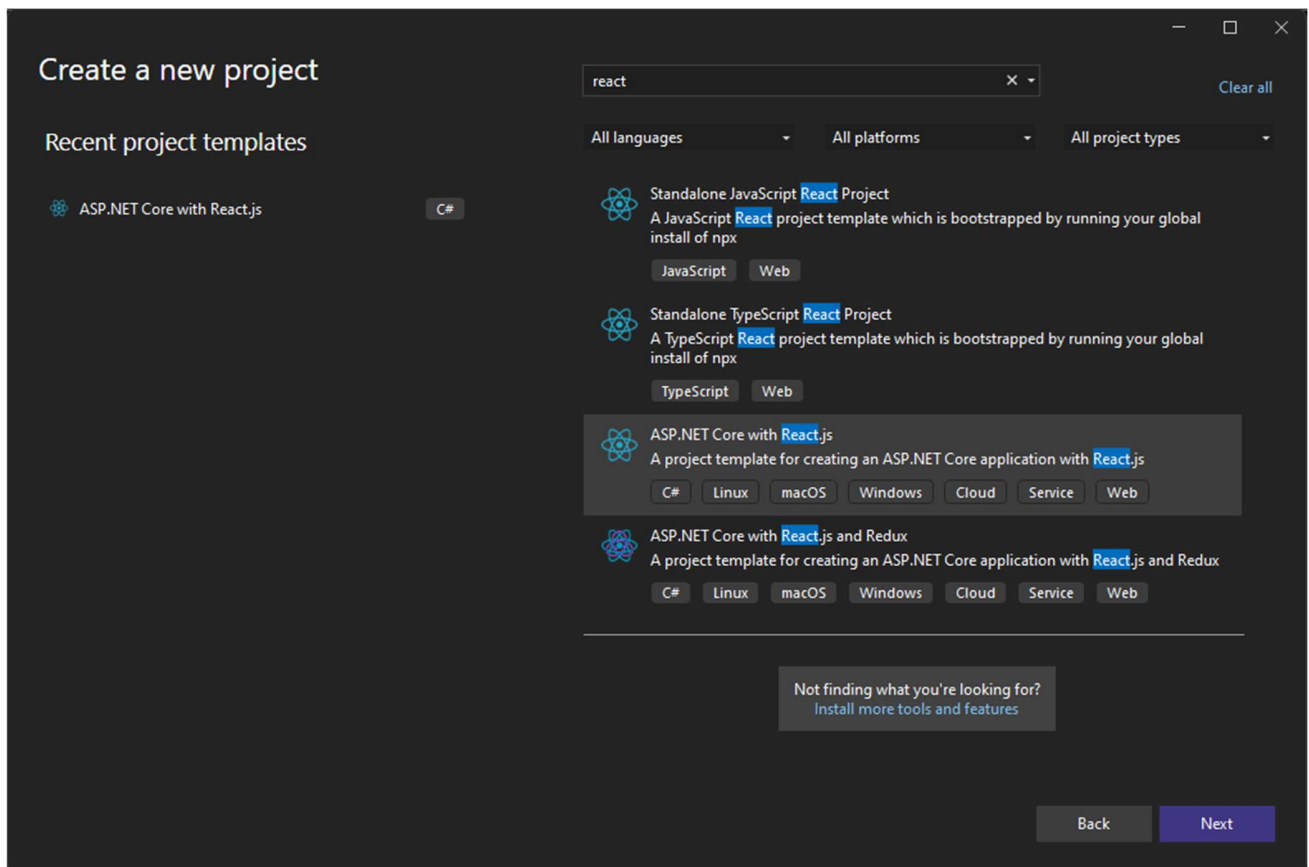


Рисунок 2.8 – Створення проекту

Відповідно до документації OpenID Connect автентифікації від Google, нам потрібно реалізувати наступні кроки для автентифікації користувача:

- Створити маркер стану для захисту від підробки
- Надіслати запит на автентифікацію в Google
- Підтвердити маркер стану захисту від підробки
- Обміняти code на маркер доступу та ідентифікатор
- Отримати інформацію про користувача з ідентифікаційного токена
- Автентифікувати користувача

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				15
Змн.	Арк.	№ докум.	Підпис	Дата		

Отже, реалізуємо створення маркеру для захисту від підробки. Це має бути унікальний рядок, тому використаємо для цього GUID:

```
private const string PkceSessionKey = "codeVerifier";

public string GenerateCodeVerifier()
{
    var codeVerifier = Guid.NewGuid().ToString();

    SetCodeVerifier(codeVerifier);

    return hashHelper.ComputeHash(codeVerifier);
}

public string? GetCodeVerifier()
    => httpContextAccessor.HttpContext?.Session.GetString(PkceSessionKey);

public void SetCodeVerifier(string codeVerifier)
    => httpContextAccessor.HttpContext?.Session.SetString(PkceSessionKey,
codeVerifier);
```

Реалізуємо метод, який створить URI автентифікації OpenID Connect, на який потрібно перенаправити користувача для автентифікації в Google:

```
public string CreateAuthenticationRequestUrl(bool isRegistration = false)
{
    var redirectUri = isRegistration
        ? googleOAuthConfiguration.RegistrationRedirectUri
        : googleOAuthConfiguration.LoginRedirectUri;

    if (string.IsNullOrEmpty(redirectUri))
        throw new ArgumentNullException("Google OAuth provider redirect URLs are not
configured.");

    var clientId = googleOAuthConfiguration.ClientId;
    if (string.IsNullOrEmpty(clientId))
        throw new ArgumentNullException("The ClientId credential of Google OAuth
server are not configured.");

    var codeChallenge = codeVerificationHelper.GenerateCodeVerifier();
    var queryString = new Dictionary<string, string?>
    {
        { "client_id", googleOAuthConfiguration.ClientId },
        { "redirect_uri", redirectUri },
        { "response_type", "code" },
        { "scope", "openid profile email" },
        { "code_challenge", codeChallenge },
        { "code_challenge_method", "S256" },
```

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				16
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        { "access_type", "offline" }
    };

    return QueryHelpers.AddQueryString(AuthorizationEndpoint, queryString);
}

```

У відповідь ми отримаємо code, одноразовий код авторизації, який сервер може обміняти на маркер доступу та ідентифікатор за допомогою POST запиту на провайдер аутентифікації, тобто Google. Даний запит має містити створений раніше маркер для захисту від підробки, щоб провайдер міг перевірити його валідність:

```

public async Task<TokenInfo?> ExchangeAuthorizationCode(string code, bool
isRegistration = false)
{
    var redirectUri = isRegistration
        ? googleOAuthConfiguration.RegistrationRedirectUri
        : googleOAuthConfiguration.LoginRedirectUri;

    var clientId = googleOAuthConfiguration.ClientId;
    var clientSecret = googleOAuthConfiguration.ClientSecret;

    if (string.IsNullOrEmpty(redirectUri) || string.IsNullOrEmpty(clientId) ||
string.IsNullOrEmpty(clientSecret))
        throw new ArgumentNullException("One or more Google OAuth provider
credentials are not configured.");

    var codeVerifier = codeVerificationHelper.GetCodeVerifier();
    if (string.IsNullOrEmpty(codeVerifier))
        return null;

    var bodyParams = new Dictionary<string, string>
    {
        { "client_id", clientId },
        { "client_secret", clientSecret },
        { "code", code },
        { "grant_type", "authorization_code" },
        { "redirect_uri", redirectUri },
        { "code_verifier", codeVerifier }
    };

    return await httpRequestHelper.SendPostRequest<TokenInfo>(TokenEndpoint,
bodyParams);
}

```

У відповідь на запит заміни коду ми отримаємо JSON зі значеннями id_token, access_token, expires_in, refresh_token, scope та token_type. Значення access_token нам

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				17
Змн.	Арк.	№ докум.	Підпис	Дата		

потрібне для роботи з Google API, наприклад, змінити опис каналу на YouTube, отримати інформацію профілю та безліч інших дій з продуктами Google, на які погодився користувач на екрані згоди. Ми налаштували лише отримання даних профіля, тому реалізуємо саме цю дію:

```
public async Task<UserInfo?> GetUserInfo(string accessToken)
{
    var queryParams = new Dictionary<string, string?>
    {
        { "access_token", accessToken }
    };

    return await httpRequestHelper.SendGetRequest<UserInfo>(UserInfoEndpoint,
queryParams);
}
```

Значення id_token ми будемо використовувати для аутентифікації, але він має обмежений термін дії (як і access_token) і для його оновлення потрібно відправити запит з відповідними параметрами:

```
public async Task<TokenInfo?> RefreshToken(string refreshToken)
{
    if (string.IsNullOrEmpty(googleOAuthConfiguration.ClientId) ||
string.IsNullOrEmpty(googleOAuthConfiguration.ClientSecret))
        throw new ArgumentNullException("Client credentials of Google OAuth server
are not configured.");

    var bodyParams = new Dictionary<string, string>
    {
        { "client_id", googleOAuthConfiguration.ClientId },
        { "client_secret", googleOAuthConfiguration.ClientSecret },
        { "grant_type", "refresh_token" },
        { "refresh_token", refreshToken }
    };

    return await httpRequestHelper.SendPostRequest<TokenInfo>(TokenEndpoint,
bodyParams);
}
```

У відповідь ми отримаємо нову пару id_token та access_token. З access_token ми не можемо отримати жодної інформації, він підписаний секретним ключем і використовується лише Google API, а id_token має публічну частину, з якої ми можемо отримати ідентифікатор та інформацію користувача, термін дії унікальний ідентифікатор токена на інші поля, які використовуються для валідації.

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				18
Змн.	Арк.	№ докум.	Підпис	Дата		

На даному етапі можна вважати процес аутентифікації завершеним. У нас є все, щоб, отримавши інформацію про користувача з токена, зробити запит до бази даних користувачів програми. Якщо користувач уже існує у нашій базі даних, ми повинні почати сеанс програми для цього користувача, якщо відповідь Google API відповідає всім вимогам входу.

Може виникнути ситуація, коли користувач припинить взаємодію з нашим додатком та захоче припинити доступ до його профілю. Для цього він може зробити це вручну у налаштуваннях свого профіля Google, але розробникам додатку варто автоматизувати даний процес:

```
public async Task<bool> RevokeToken(string token)
{
    var queryParams = new Dictionary<string, string?>
    {
        { "token", token }
    };

    var response = await
HttpRequestHelper.SendHttpRequestAsync<object>(HttpMethod.Post, RevocationEndpoint,
queryParams: queryParams);
    return response != null;
}
```

Після виконання такого запиту, всі id_token, access_token та refresh_token стають недійсними.

Перевіримо роботу нашого функціоналу на практиці. Отримаємо URI аутентифікації:

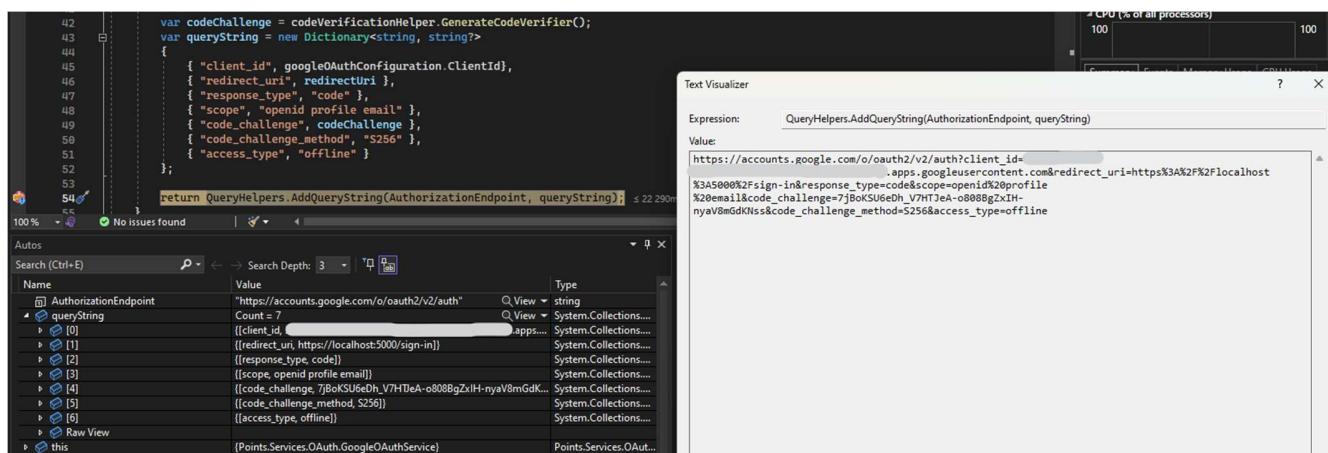


Рисунок 2.9 – Отриманий URI аутентифікації

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				19
Змн.	Арк.	№ докум.	Підпис	Дата		

Отже, був сформований URI з необхідними параметрами та маркер для захисту від підробки yJWzqbU7hTlyJ9I0c0Esn9VQ-9n1wegSpjgjfWyBOjA.

Оберемо наш обліковий запис:

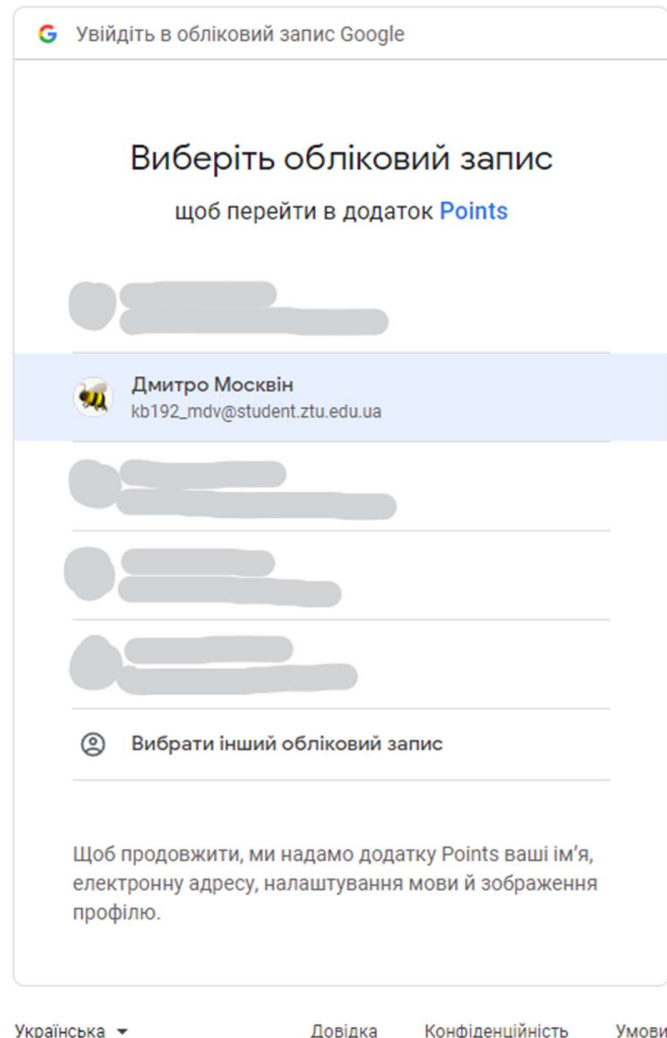


Рисунок 2.10 – Екран згоди користувача

Після підтвердження нас буде перенаправлено на redirect URI, який ми вказали у параметрах запиту, з кодом та іншою інформацією у параметрах посилання:

<https://localhost:5000/sign-in?code=4/0AWtgzh4e7-bgycyTetNaExcMzQMDSaldAG161ktfbBoZQYRYXALZ-BaYXMOMraHKtXaR2g&scope=email%20profile%20openid%20https://www.googleapis.com/auth/userinfo.profile%20https://www.googleapis.com/auth/userinfo.email&authuser=1&hd=student.ztu.edu.ua&prompt=none>.

		Москвін Д. В.			«Житомирська політехніка». 23.125.09.000 – ПР	Арк.
		Окунькова О.О.				20
Змн.	Арк.	№ докум.	Підпис	Дата		

Обмінємо отриманий код на токени:

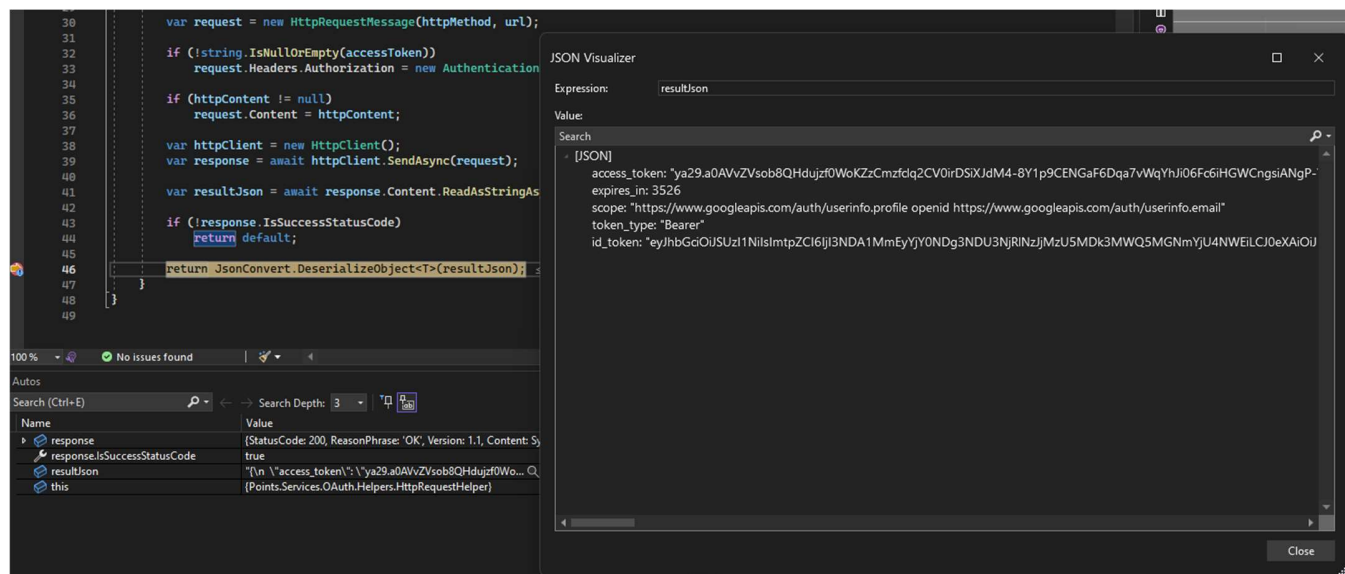


Рисунок 2.11 – Отримані токени в обмін на код

Тепер спробуємо отримати дані з Google API за допомогою токену доступу:

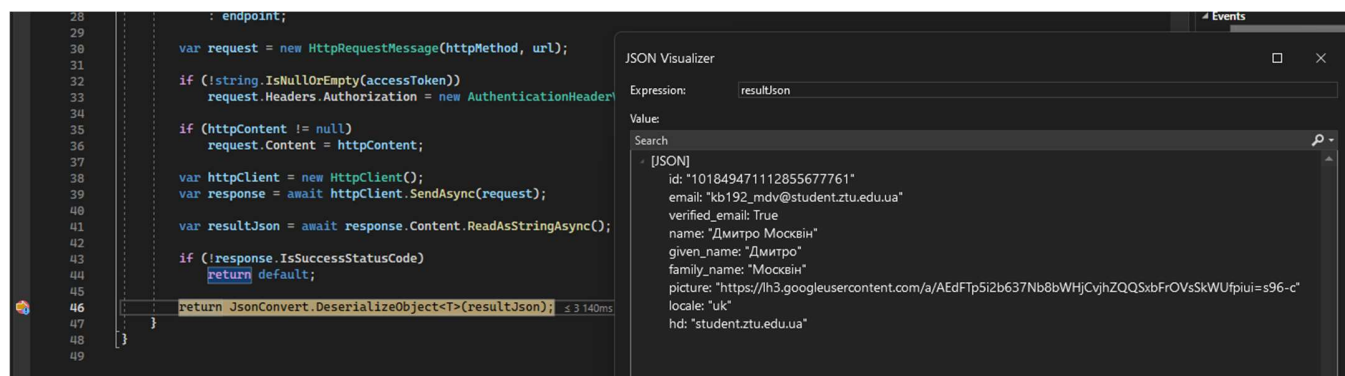


Рисунок 2.12 – Отримані дані профіля з Google API

Оновимо отримані токени:

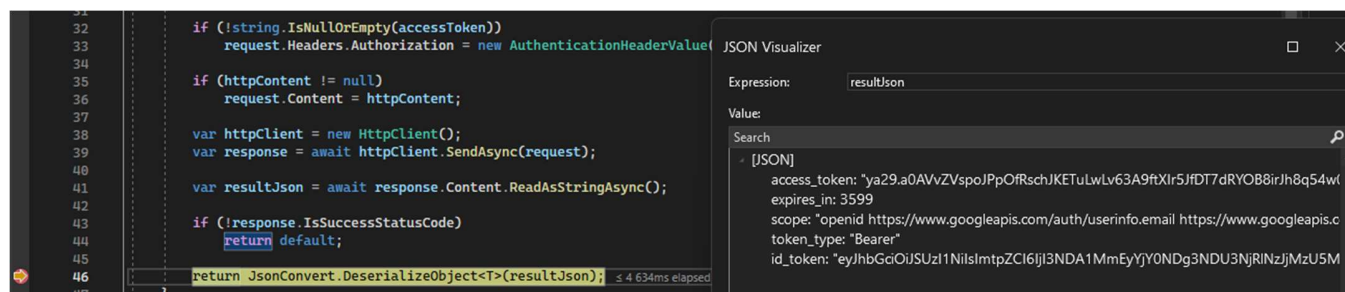


Рисунок 2.13 – Результат оновлення токенів

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				21
Змн.	Арк.	№ докум.	Підпис	Дата		

Отже, ми отримали новий токен доступу та ідентифікації, а також нове значення терміну їх дії. Варто зазначити, що у даному списку немає refresh токена, оскільки він видається лише від час аутентифікації та має вічний термін дії.

Перейдемо у налаштування облікового запису Google та переглянемо сервіси, яким ми надали доступ:

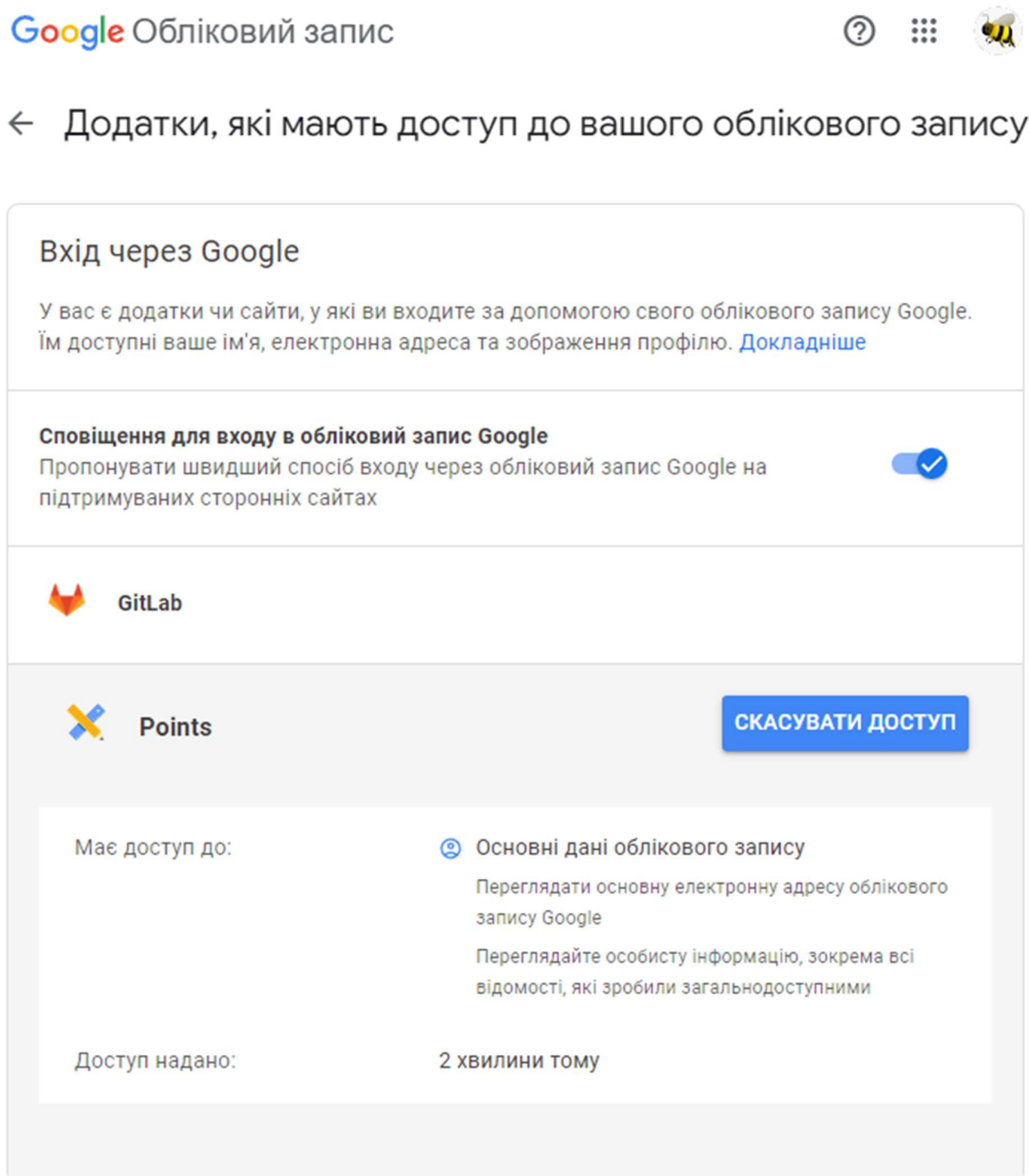


Рисунок 2.14 – Додатки, які мають доступ до вашого облікового запису

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				22
Змн.	Арк.	№ докум.	Підпис	Дата		

Отже, бачимо у списку наш додаток. Надішлемо запит на відміну доступу та перевіримо зміни:

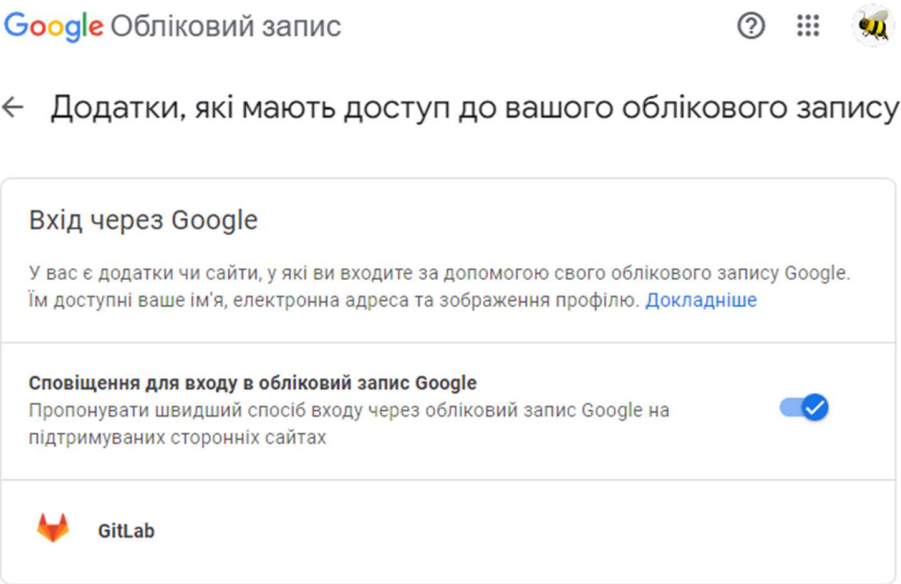


Рисунок 2.15 – Додатки, які мають доступ до вашого облікового запису

Отже, ми успішно відмінили доступ нашого додатку до облікового запису користувача.

Спробуємо відмінити доступ вручну та оновити токени:

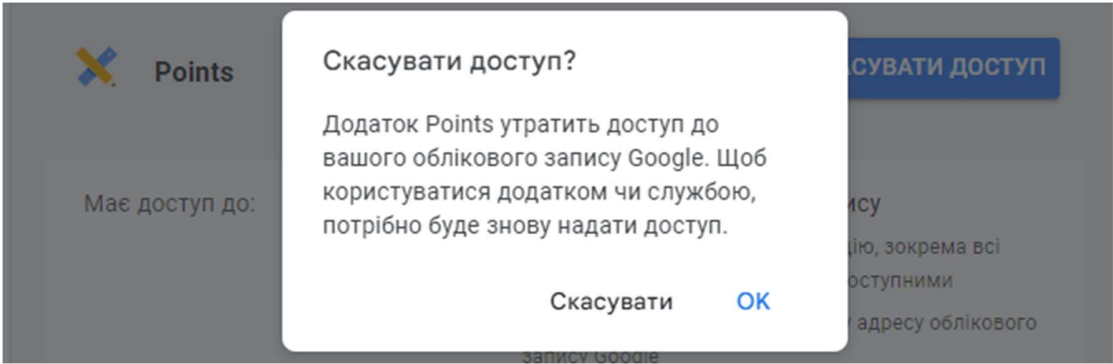


Рисунок 2.16 – Скасування доступу

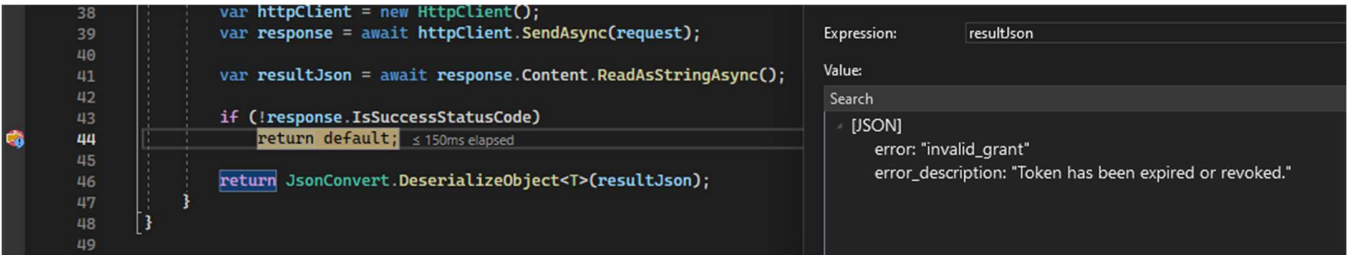


Рисунок 2.17 – Результат спроби оновити токени

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				23
Змн.	Арк.	№ докум.	Підпис	Дата		

Отже, ми успішно пройшли повний шлях, описаний в документації Google та можемо зробити висновок, що поставлене завдання виконано.

Завдання 3. Задеплоїти створений веб-додаток за допомогою сервісу Microsoft Azure.

Створимо ресурс WebApp:

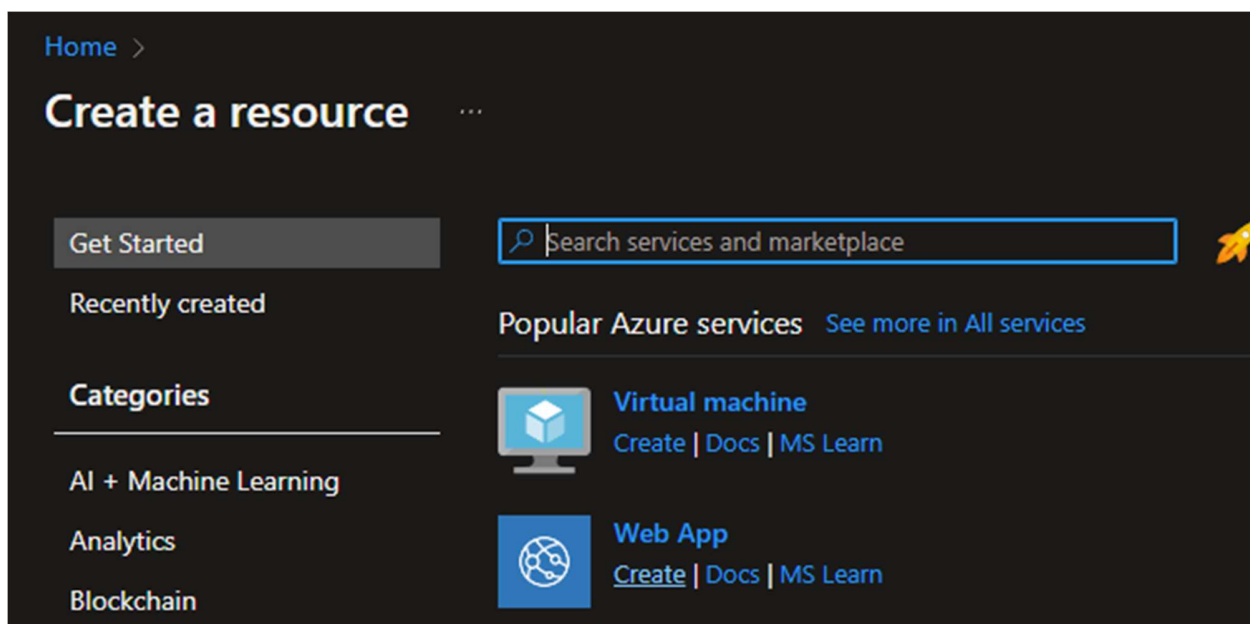


Рисунок 2.18 – Створення ресурсу

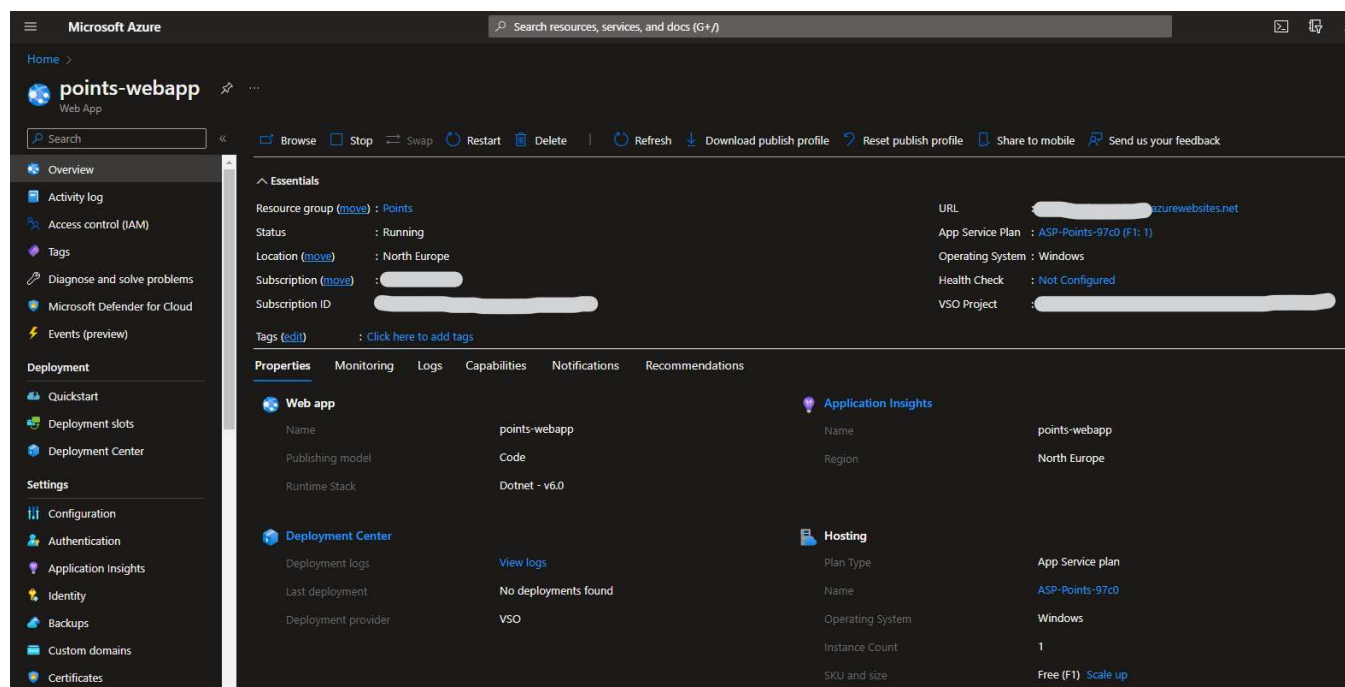


Рисунок 2.19 – Створений ресурс WebApp

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		24

Перейдемо до Deployment Center та налагодимо заздалегідь створений репозиторій у Azure DevOps як джерело:

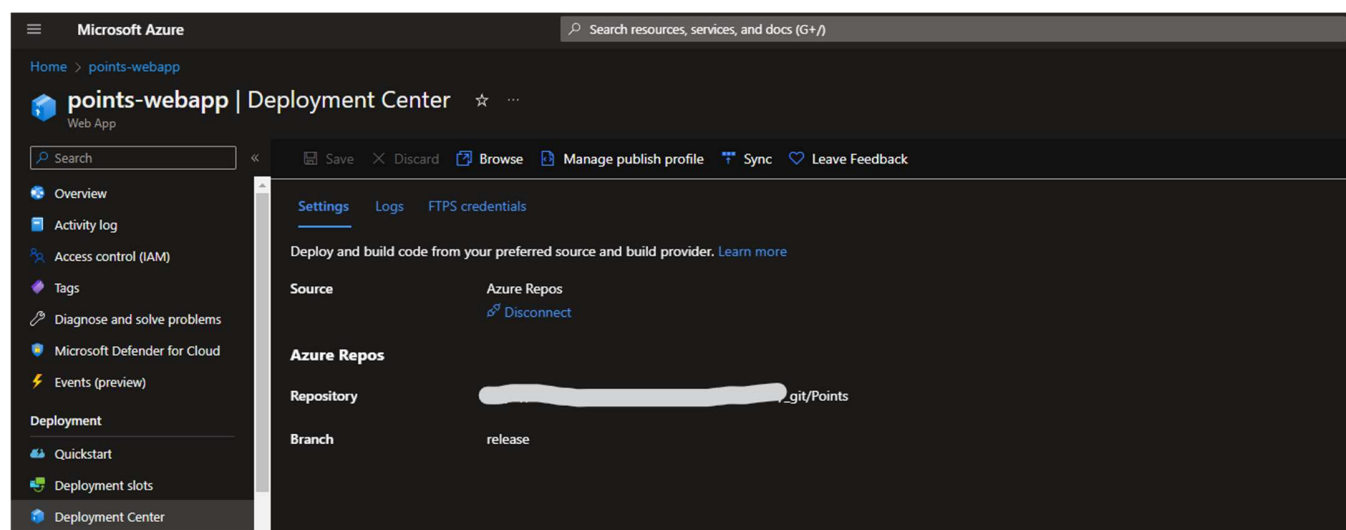


Рисунок 2.20 – Налаштування джерела

Тепер всі внесені в гілку release зміни будуть автоматично деплоїтись.

Додамо наш URI в облікові дані проекту в Google API Console:

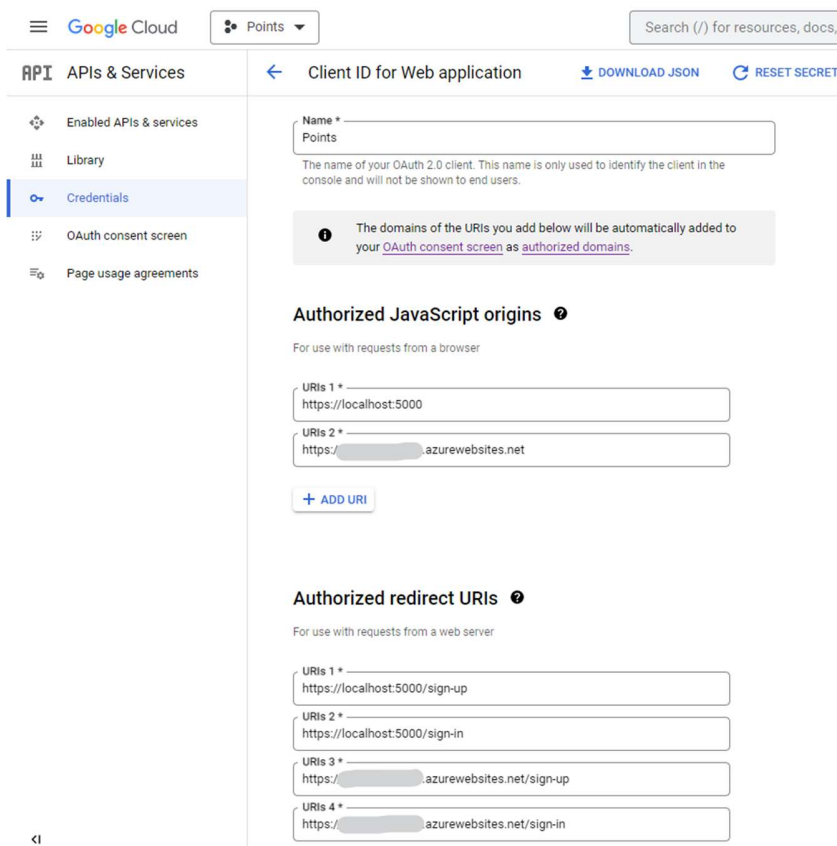


Рисунок 2.21 – Оновлені облікові дані проекту в Google API Console

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				25
Змн.	Арк.	№ докум.	Підпис	Дата		

Перейдемо за налагодженим посиланням та перевіримо дієздатність додатку:

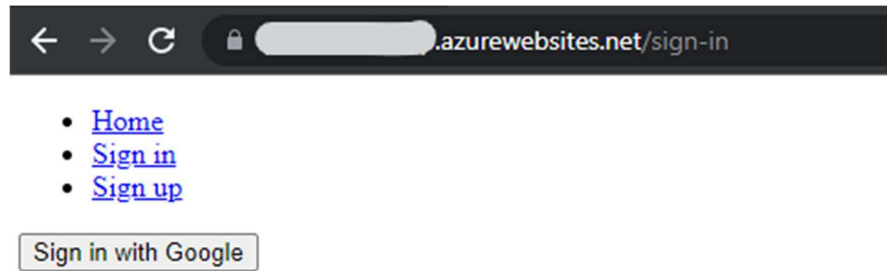


Рисунок 2.22 – Задеплоїний веб-додаток

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				26
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

Було опрацьовано теоретичну частину та на основі отриманих знань виконано практичні завдання, а саме, налаштовано проект у Google API Console: налагоджено URI перенаправлення, щоб отримати облікові дані, та інформацію на екрані згоди користувача; реалізовано процес аутентифікації на основі технології єдиного входу відповідно до документації Google Identity; перевірено коректність роботи розробленого веб-додатку та виконано його опублікування за допомогою сервісу Microsoft Azure.

		Москвін Д. В.			«Житомирська політехніка».23.125.09.000 – ПР	Арк.
		Окунькова О.О.				27
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Single Sign-On - Auth0 [Електронний ресурс] – Режим доступу до ресурсу: <https://auth0.com/docs/authenticate/single-sign-on>.
2. OAuth 2.0 [Електронний ресурс] – Режим доступу до ресурсу: <https://oauth.net/2/>.
3. Best SSO Providers And Software for all Applications [Електронний ресурс] – Режим доступу до ресурсу: <https://www.getkisi.com/blog/best-single-sign-on-sso-providers>.
4. Top Single Sign-On (SSO) Solutions for 2022 [Електронний ресурс] – Режим доступу до ресурсу: <https://www.esecurityplanet.com/products/single-sign-on-solutions/>.
5. OpenID Connect Google Identity Authentication [Електронний ресурс] / Google – Режим доступу до ресурсу: <https://developers.google.com/identity/openid-connect/openid-connect>.
6. Using OAuth 2.0 for Web Server Applications [Електронний ресурс] – Режим доступу до ресурсу: <https://developers.google.com/identity/protocols/oauth2/web-server>.
7. Revoking a token [Електронний ресурс] – Режим доступу до ресурсу: <https://developers.google.com/identity/protocols/oauth2/web-server#tokenrevoke>.
8. OAuth 2.0 Scopes for Google APIs [Електронний ресурс] – Режим доступу до ресурсу: <https://developers.google.com/identity/protocols/oauth2/scopes>.
9. OAuth 2.0 Policies [Електронний ресурс] – Режим доступу до ресурсу: <https://developers.google.com/identity/protocols/oauth2/policies>.
10. Authenticating the user [Електронний ресурс] – Режим доступу до ресурсу: <https://developers.google.com/identity/openid-connect/openid-connect#authenticatingtheuser>.

		Москвін Д. В.			«Житомирська політехніка». 23.125.09.000 – ПР	Арк.
		Окунькова О.О.				28
Змн.	Арк.	№ докум.	Підпис	Дата		