Лабораторна робота №2

Звіт

З дисципліни “Хмарні технології ”

на тему: “Microsoft Azure Entra ID”.

Студента 3 курсу: Групи МІТ-31 Демиденко Андрій

**Київ - 2024р.**

**Комплексний звіт**

**Мета:** . Організація автентифікації та авторизації користувачів за допомогою Microsoft Azure Entra ID.

**Хід роботи**

Для початку потрібно Зареєструвати програму на порталі Azure. Прейдемо на портал Azure за адресою https://portal.azure.com/.Потрібно переконайтеся, що у є активна підписка або хоча б пробна версія. Знаходимо на порталі «Microsoft Entra ID» . Натискаємо «Керувати», а потім «Реєстрація програми», після цього «Нова реєстрація», щоб зареєструвати нову програму (рис. 2.1). Вказуємо ім’я для нової програми та, що важливо, вибрати «підтримувані типи облікових записів» відповідно до вимог, додаємо URI перенаправлення як адресу локального хосту і зареєструємо цю адресу локального хосту. наприклад, для мене це https://localhost:7267/ (запущений локальний хост може бути іншим) і натискаємо «Зареєструватись» (рис. 2.2).

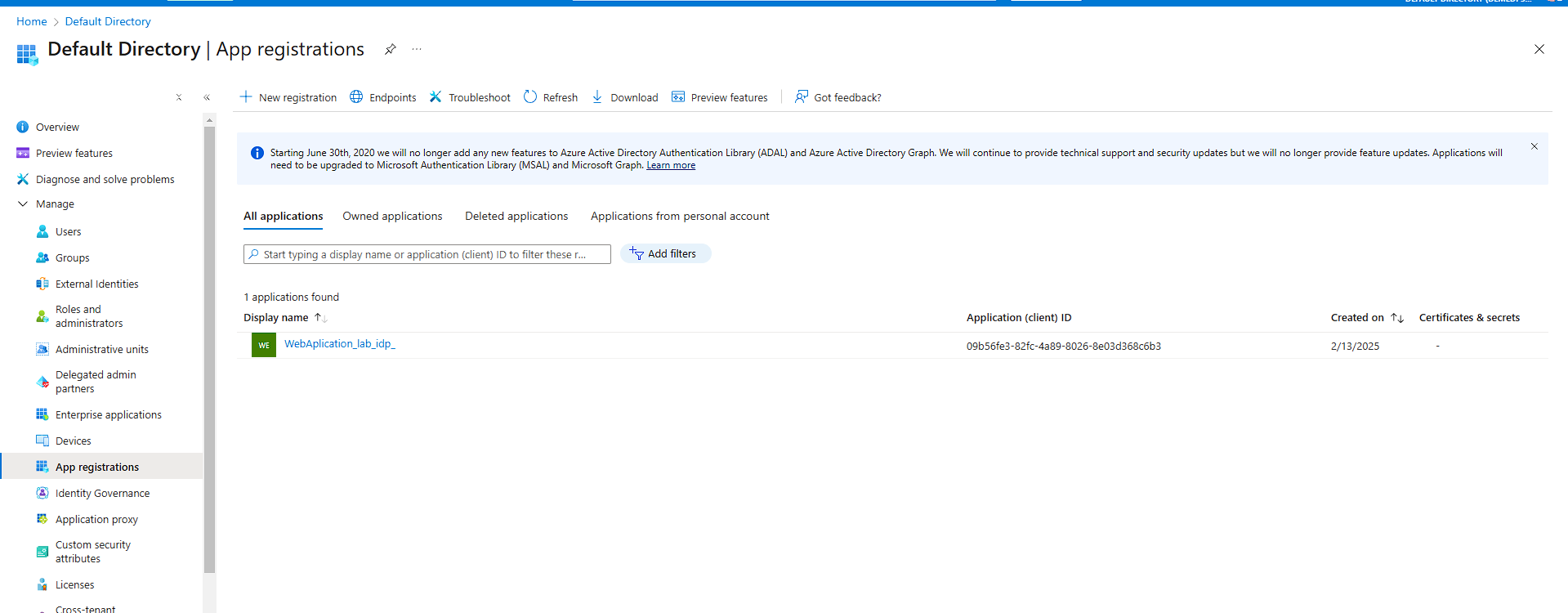


Рис. 2.1 – Шлях до реєстрації додатку.

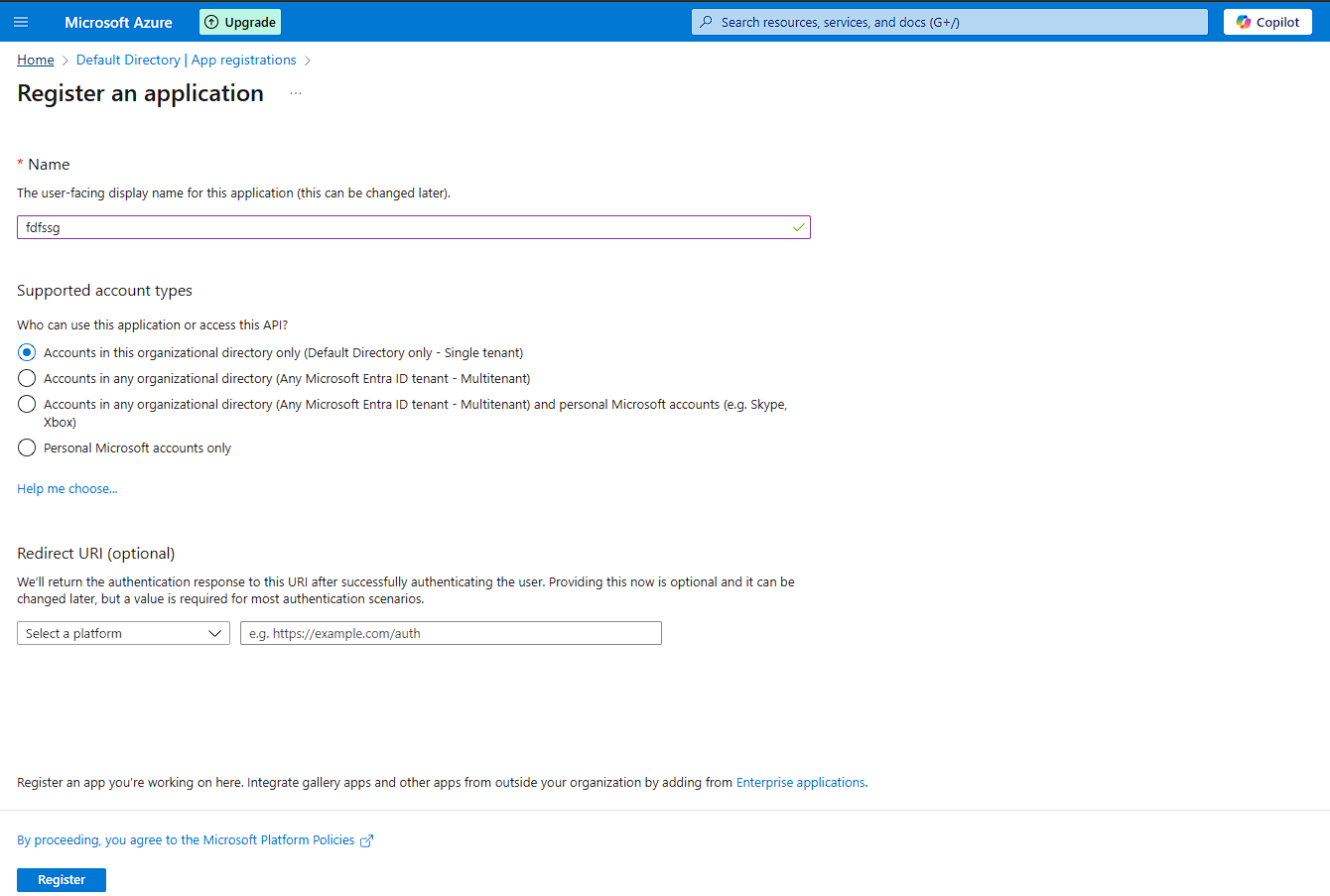


Рис. 2.2 – Реєстрація додатку.

Після реєстрації програми можна побачити основні ідентифікатори програми, як-от ідентифікатор каталогу (клієнта) та ідентифікатор програми (клієнта) тощо, які нам потрібні для визначення цих облікових даних у файлі appsettings.json.

Додаємо наведений нижче код у файл appsettings.json основної програми .net (рис. 2.3).



рис. 2.3 – Додаємо код до appsettings.json.

Примітка: «CallbackPath»: «/signin-oidc», цей шлях зворотного виклику важливий, коли доведеться зареєструвати URI перенаправлення в ідентифікаторі входу. рекомендується використовувати той самий шлях зворотного виклику.

Пояснення вищесказаного:

У розділі «AzureAd» містяться:

«Екземпляр»: це URL-адреса центру Microsoft Entra ID, де користувачі будуть автентифікуватися.

«ClientId»: ідентифікатор клієнта зареєстрованої програми в Microsoft Entra ID, який представляє програму.

«TenantId»: ідентифікатор клієнта Microsoft Entra ID, якому належить програма.

«CallbackPath»: це URL-адреса, на яку буде надіслано відповідь автентифікації після успішного входу.

**Додавання URI перенаправлення в Entra ID**

На порталі перейдемо до програми та оберемо «Автентифікація», а в розділі «Конфігурація платформи» «Додати платформу», а потім «Веб» і додаємо URI перенаправлення (рис. 2.4).

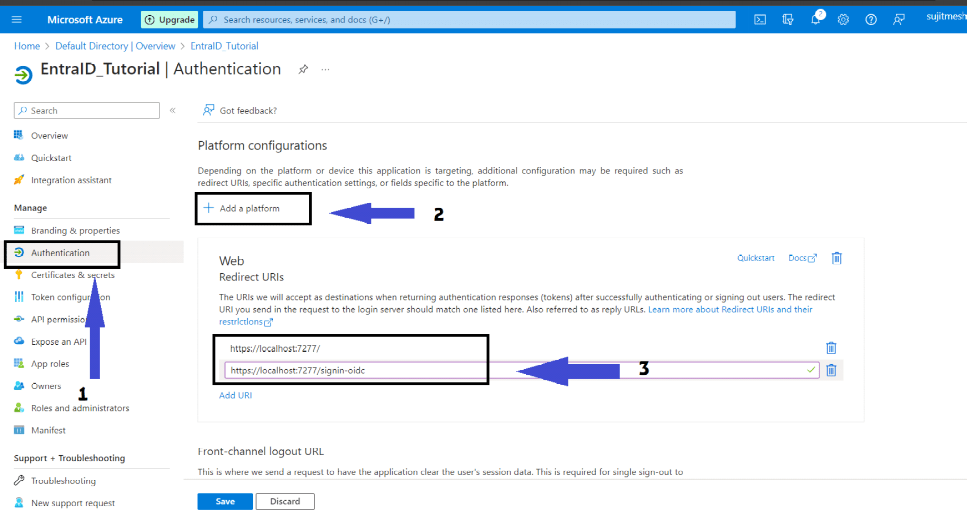


рис. 2.4 – Додаємо URI перенаправлення.

Примітка: цей «/signin-oidc», визначений як зворотний шлях у файлі appsettings.json у попередніх кроках.

Позначаємо «Маркети доступу» та «Маркети ідентифікатора» та натиснкаємо «Зберегти» (рис. 2.5).

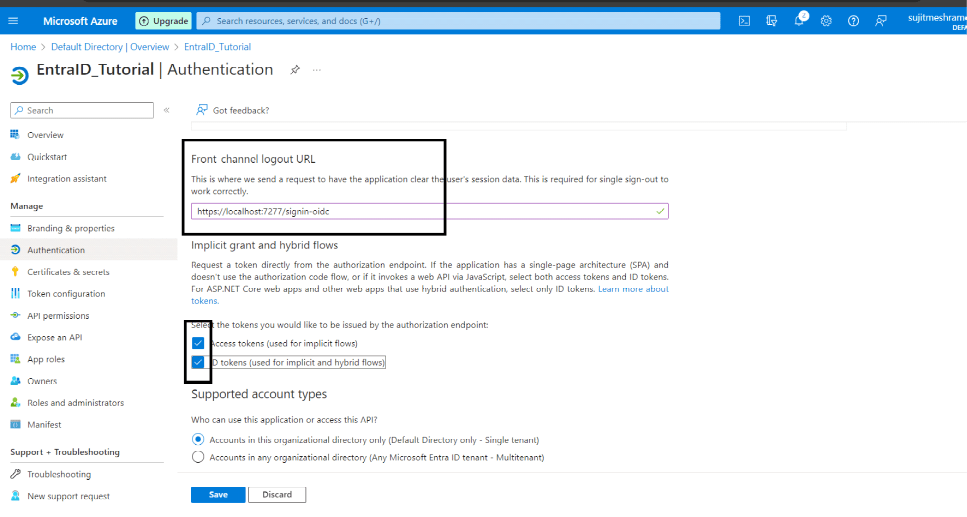
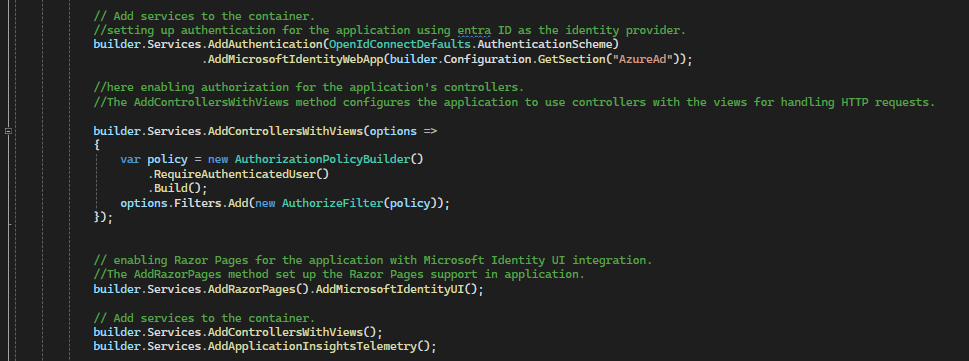


рис. 2.5 – Додаємо правильні налаштування.

Установимо такий пакет nuget у програмі: Microsoft.Identity.Web.UI та налаштуємо наступний код для program.cs. Що за що відповідає описано в коментарях коду, все було взято з документації Microsoft по Entra ID (рис. 2.6).



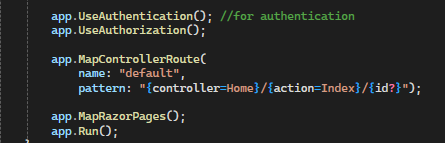


рис. 2.6 – Додаємо код у program.cs.

Додавання властивості авторизації для захисту контролерів (рис. 2.7)**.**

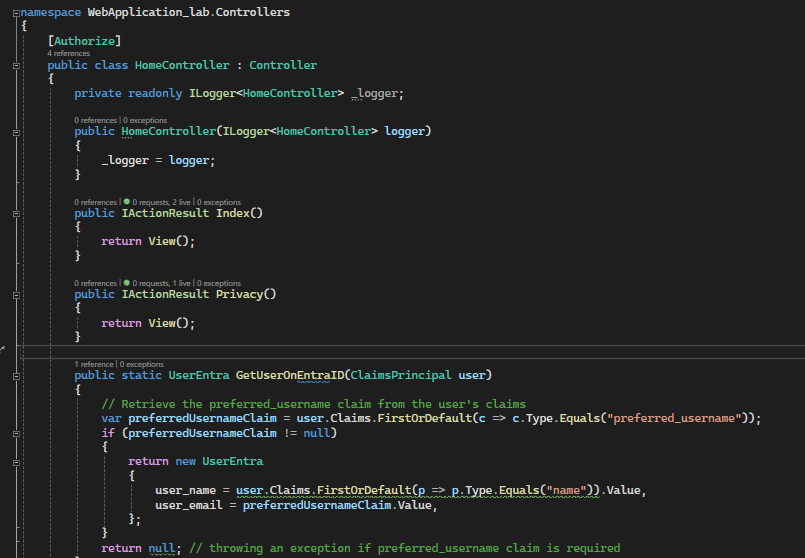
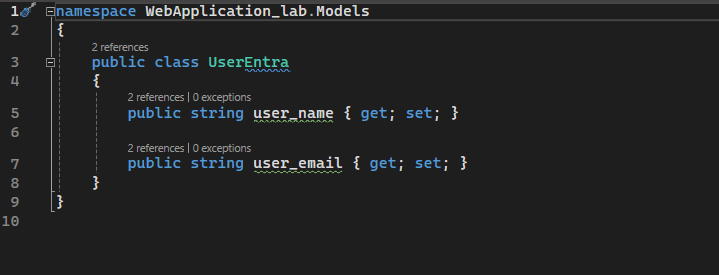
****

рис. 2.7 – Додавання властивості авторизації.

Щоб отримати деякі дані про користувача з Entra ID після автентифікації, для цього створюємо наступний клас моделі «UserEntra» (рис. 2.8)**.**

****рис. 2.8 – Створюємо клас моделі «UserEntra».

Отримання даних для вищезазначеного об’єкта за допомогою класу ClaimsPrincipal, для цього прикладу додаємо метод до класу HomeController.cs (рис. 2.8)**.**

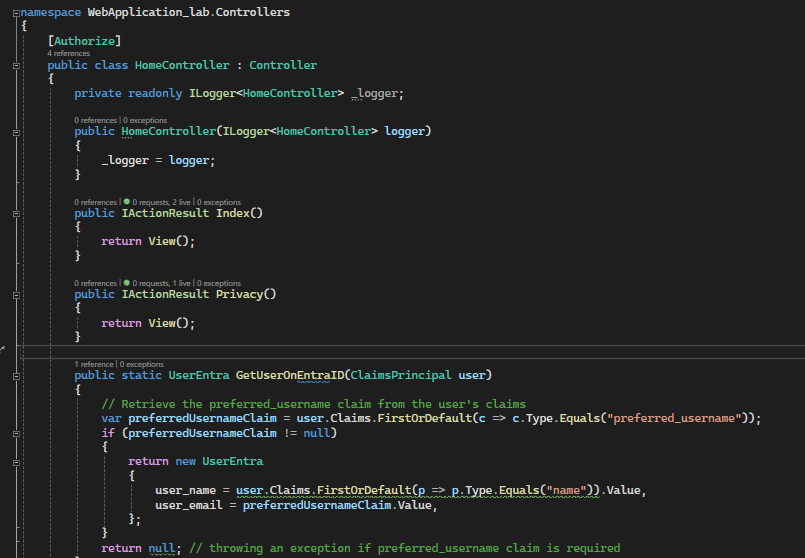
****

рис. 2.8 – Додаємо код у program.cs.

Додаємо наступний блок коду для повторного входу у файл \_Layout.cshtml (рис. 2.9).

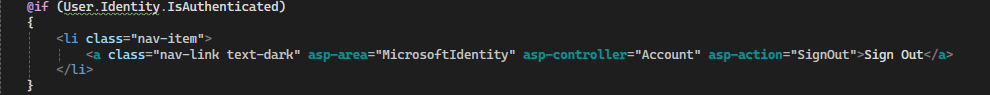


рис. 2.9 – блок коду для повторного входу \_Layout.cshtml.

Тепер запускаємо застосунок і проходимо авторизацію (рис. 2.10).

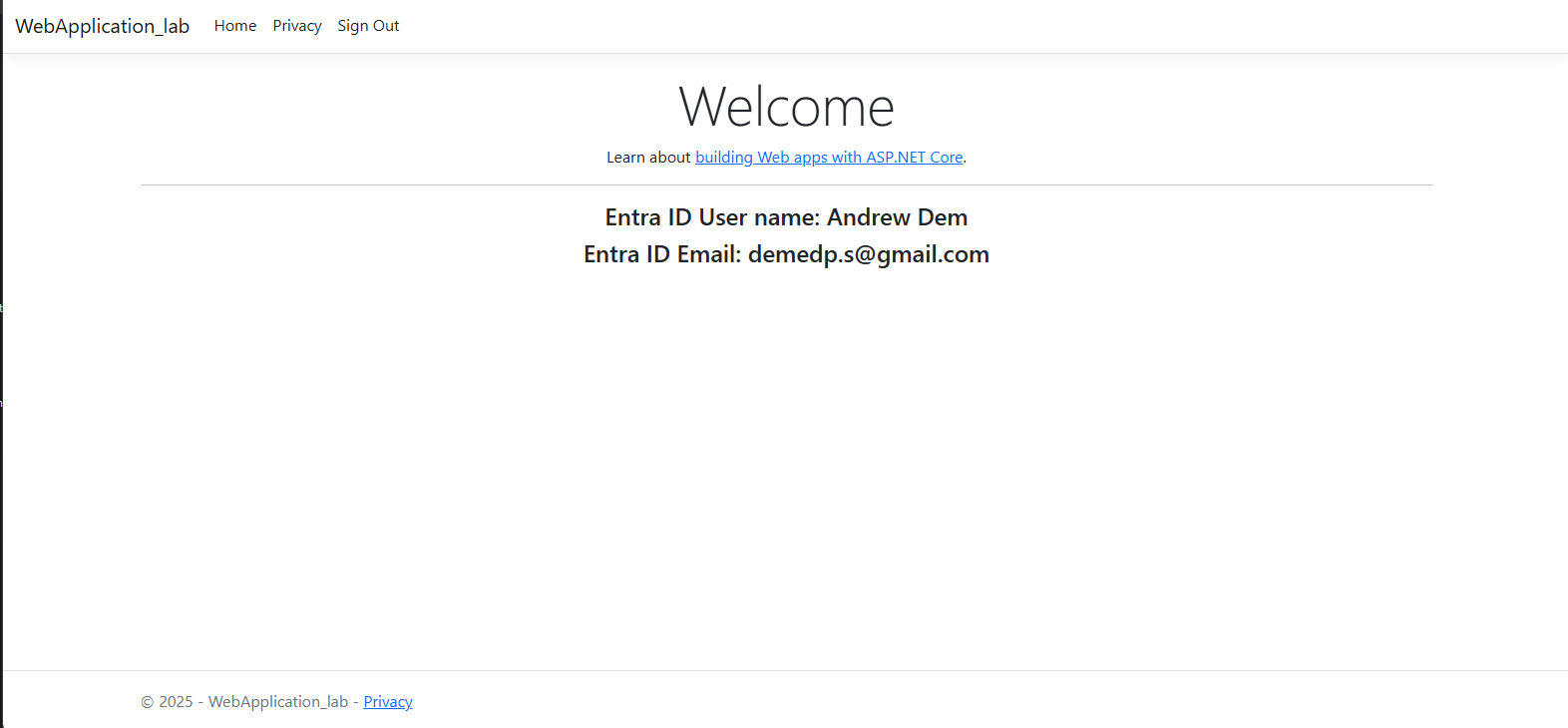
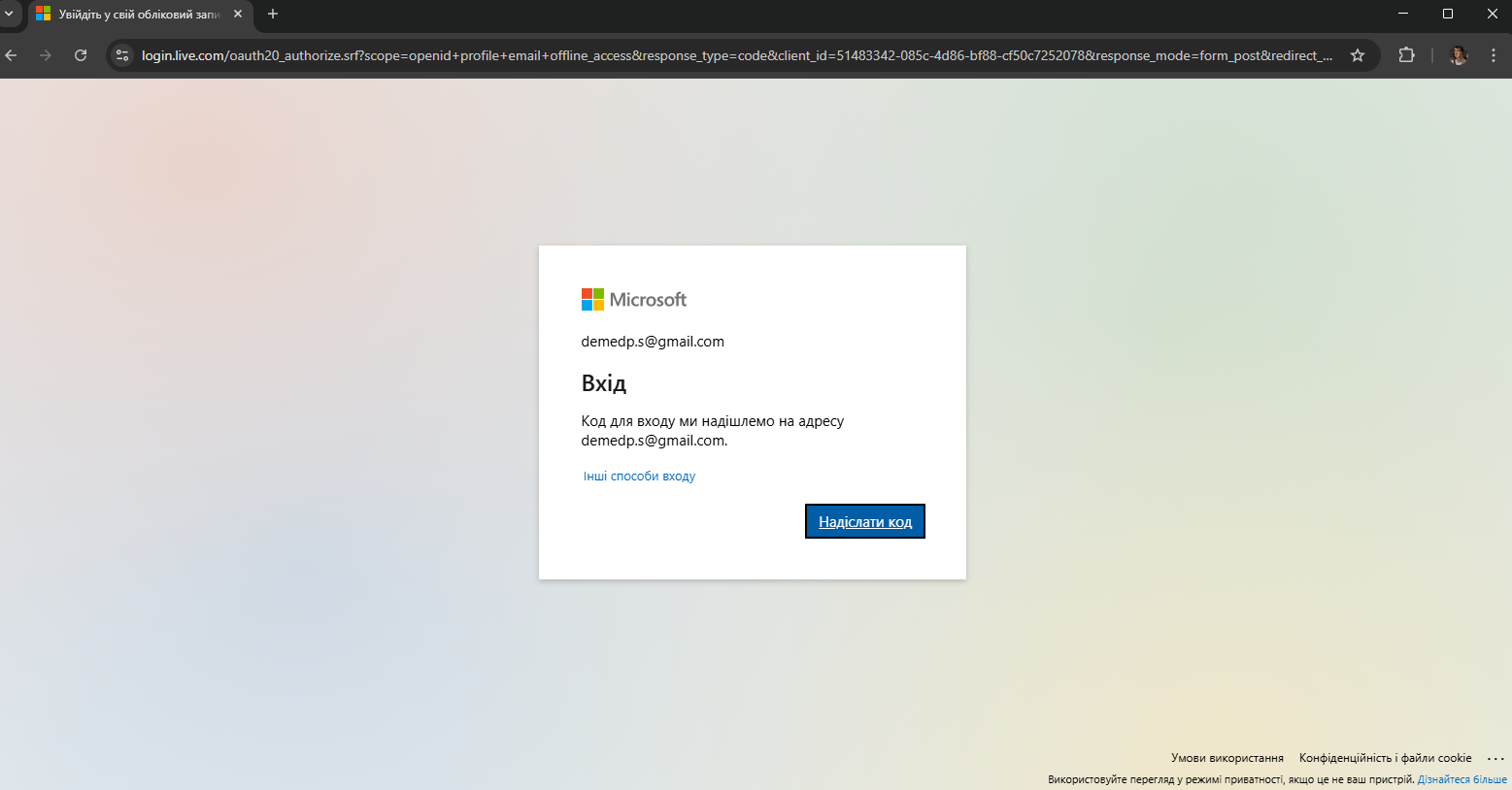


рис. 2.10 – Робоча програма.

**Висновок:** У цій роботі було розглянуто процес організації автентифікації та авторизації користувачів за допомогою Microsoft Azure Entra ID. Ми зареєстрували програму на порталі Azure, налаштували необхідні параметри, зокрема URI перенаправлення, та зберегли облікові дані у appsettings.json.

Далі було додано відповідні NuGet-пакети, налаштовано program.cs для інтеграції з Entra ID, а також захист контролерів за допомогою авторизації. Для отримання даних про користувачів було створено клас моделі та реалізовано механізм отримання інформації через ClaimsPrincipal.

[Git Reposytory](https://github.com/AndrewDem0/Cloud_Tech.git)