МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

БУДІВНИЦТВА І АРХІТЕКТУРИ

Кафедра інформаційних технологій

**Лабораторна робота № 2**

З дисципліни: Інструментальні засоби розробки програмного забезпечення

Виконав:

Студент групи ІСТм-24

Красновський Сергій

Київ-2025

**Мета:** створення базового шаблону проєкту для кросплатформного мобільного застосунку "Трекер корисних звичок" у середовищі Visual Studio 2022.

**Хід роботи**

У Visual Studio 2022 обрано шаблон **WPF App (.NET)**, назву проєкту — EchoNote, цільову платформу — **.NET 8.0**. На *Рисунку 2.1* показано встановлення потрібних компонентів у *Visual Studio Installer* (робоче навантаження **.NET desktop development**).

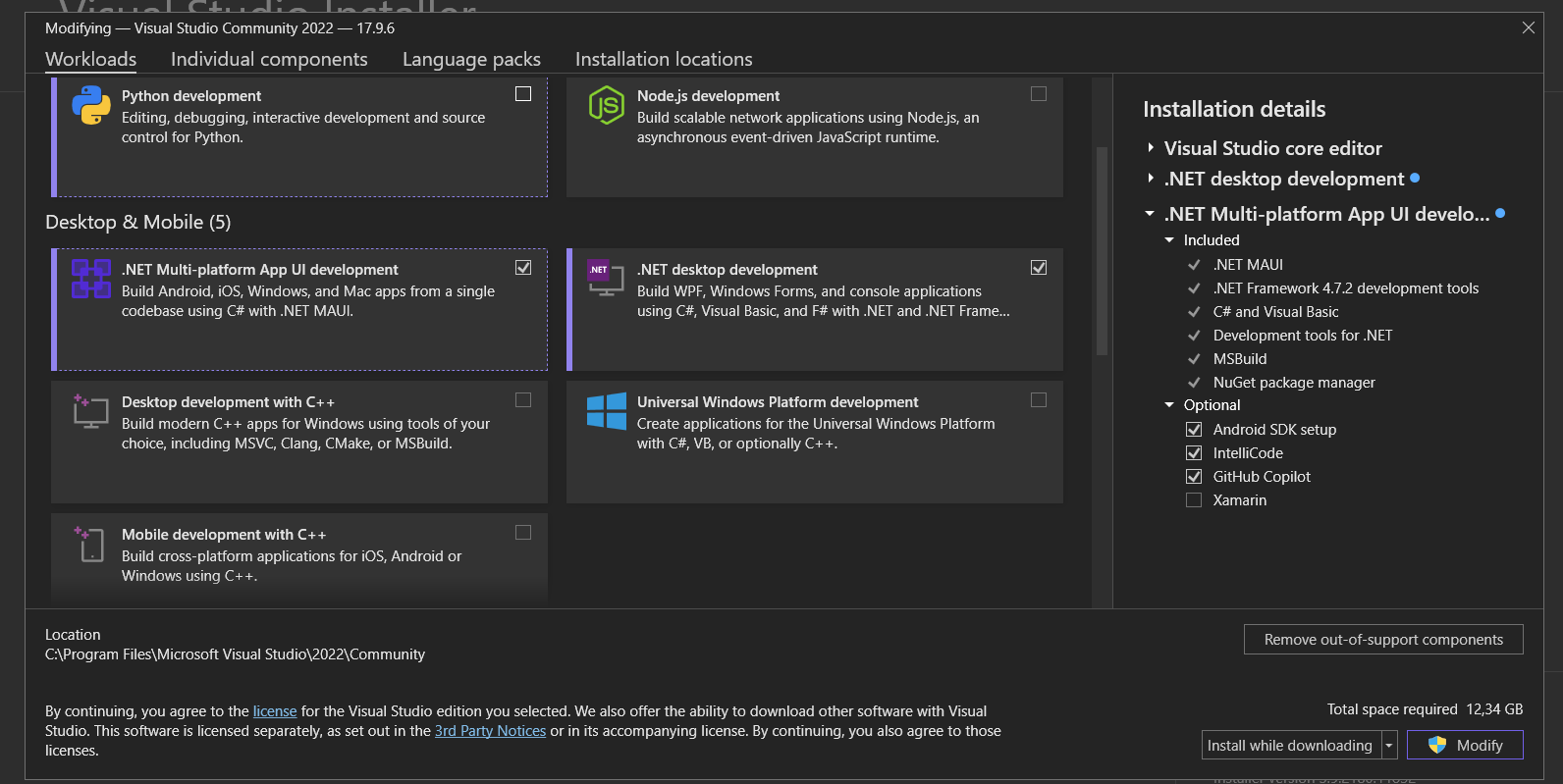
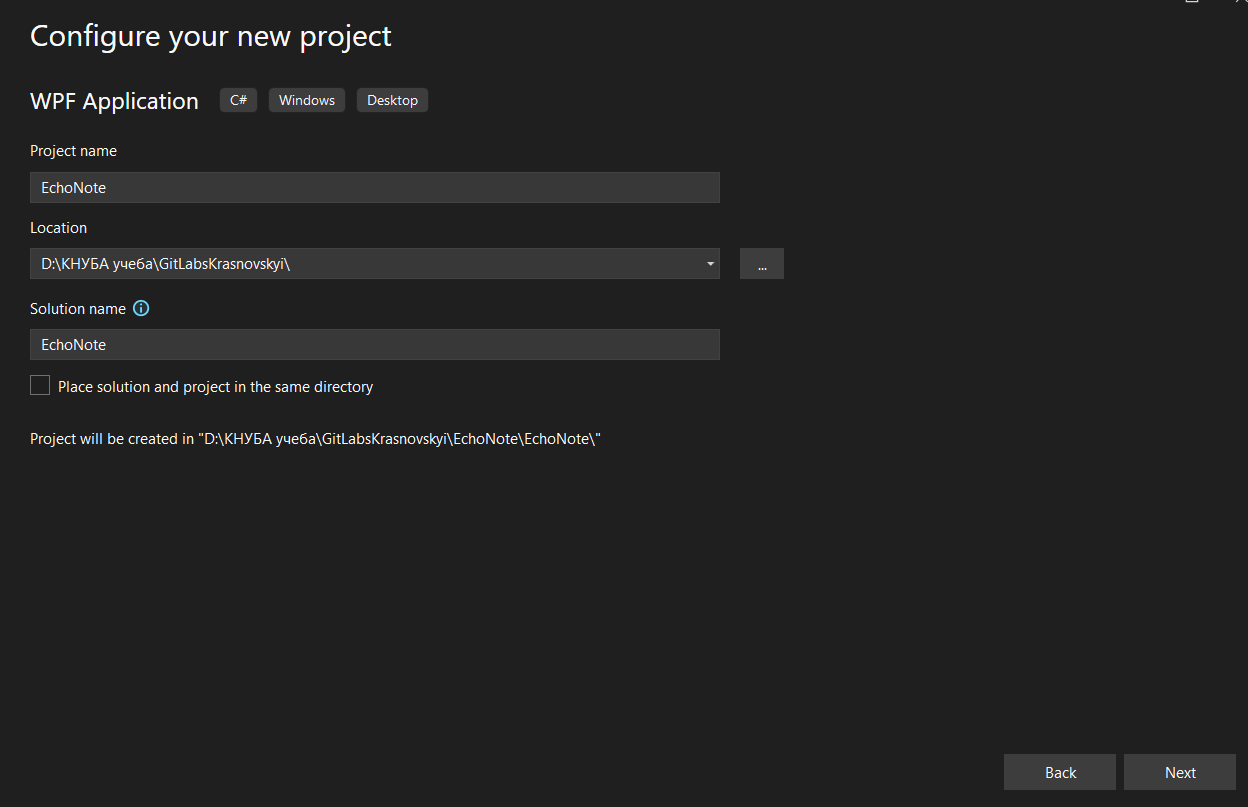


Рисунок 2.1 – Завантаження необхідних компонентів



*Рисунку 2.2* — діалог створення WPF-проєкту

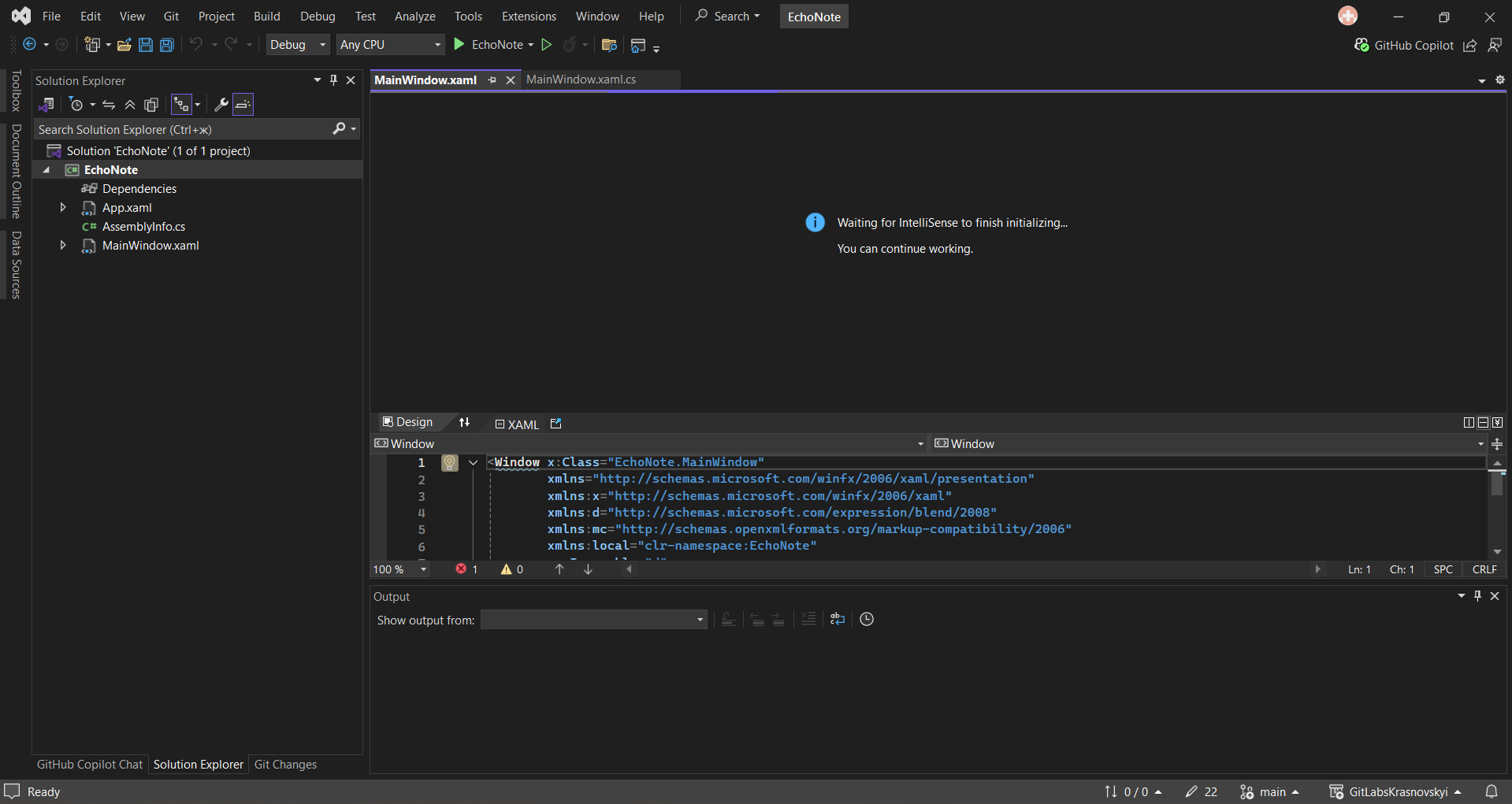


Рисунок 2.3 – Створений проект

На **Рисунку 2.2** зображено створення нового проєкту. У діалоговому вікні створення проєкту було обрано шаблон **WPF Application (C#)**, задано назву проєкту **EchoNote** та вибрано цільову платформу **.NET 8.0 (Long Term Support)**, яка є актуальною для розробки настільних застосунків на вересень 2025 року. Після натискання **Create** Visual Studio згенерувала базову структуру, включаючи файли MainWindow.xaml для користувацького інтерфейсу та App.xaml для конфігурації застосунку (**Рисунок 2.3**).

Для перевірки коректності налаштувань проєкт було скомпільовано у режимі **Debug** (*Build > Build Solution*), після чого успішно відкрилася порожня форма вікна (**Рисунок 2.4**).

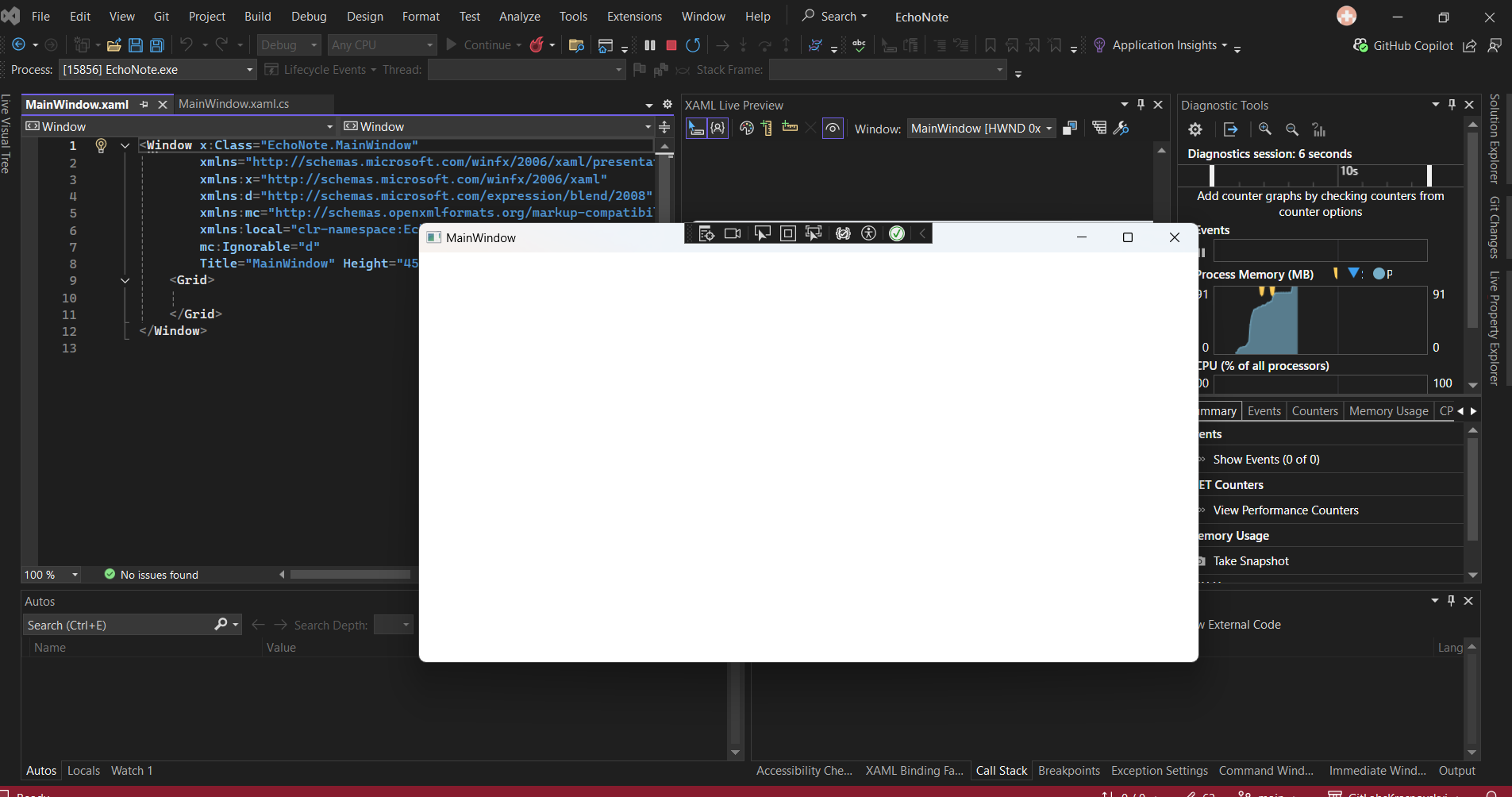


Рисунок 2.4 –порожня форма вікна

Щоб реалізувати офлайн-розпізнавання мовлення та локальне збереження транскриптів, за допомогою **NuGet Package Manager** підключено такі бібліотеки (**Рисунок 2.5**):

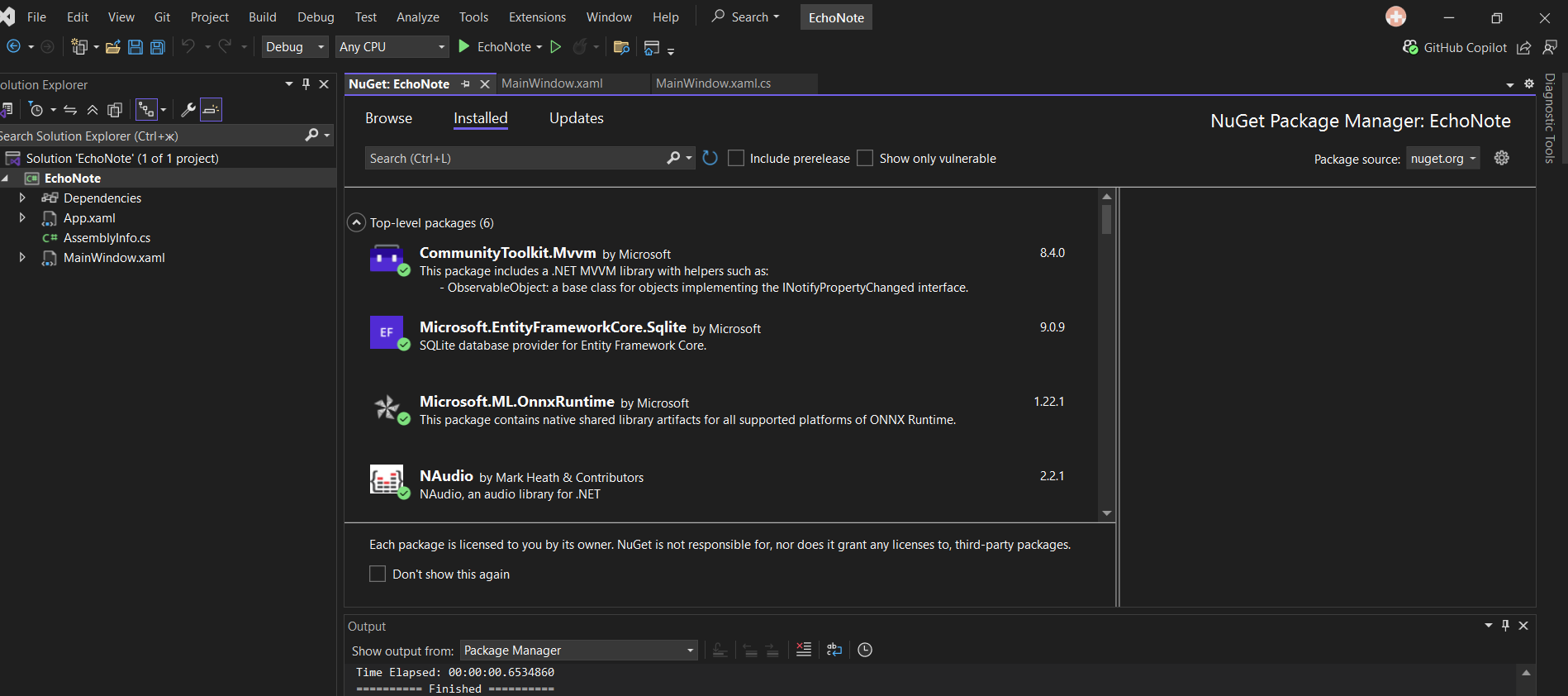


Рисунок 2.5 - NuGet Package Manager

* **Microsoft.ML.OnnxRuntime** — для запуску моделей автоматичного розпізнавання мовлення (ASR) та створення семантичних ембеддінгів повністю офлайн.
* **NAudio** — для захоплення та обробки аудіосигналу з мікрофона.
* **Microsoft.EntityFrameworkCore** та **Microsoft.EntityFrameworkCore.Sqlite** — для роботи з локальною базою даних SQLite.
* **CommunityToolkit.Mvvm** — для зручної реалізації патерну MVVM.
* **Serilog** та **Serilog.Sinks.File** — для структурованого логування подій.

Обґрунтування вибору бібліотек:

* **Microsoft.ML.OnnxRuntime** — забезпечує швидке офлайн-розпізнавання мовлення й побудову семантичних індексів без залежності від хмарних сервісів.
* **NAudio** — проста інтеграція із системним мікрофоном.
* **EF Core + SQLite** — легка й надійна база даних, оптимальна для локального зберігання транскриптів.
* **CommunityToolkit.Mvvm** — мінімізує шаблонний код при побудові інтерфейсу за MVVM.
* **Serilog** — забезпечує зручне логування під час розробки та експлуатації.

Після успішного тестового запуску файли проєкту було додано до репозиторію, створеного під час виконання першої лабораторної роботи, та за допомогою **Git** виконано коміт для ініціалізації нових файлів. Репозиторій на GitHub із першим комітом зображено на **Рисунку 2.6**.

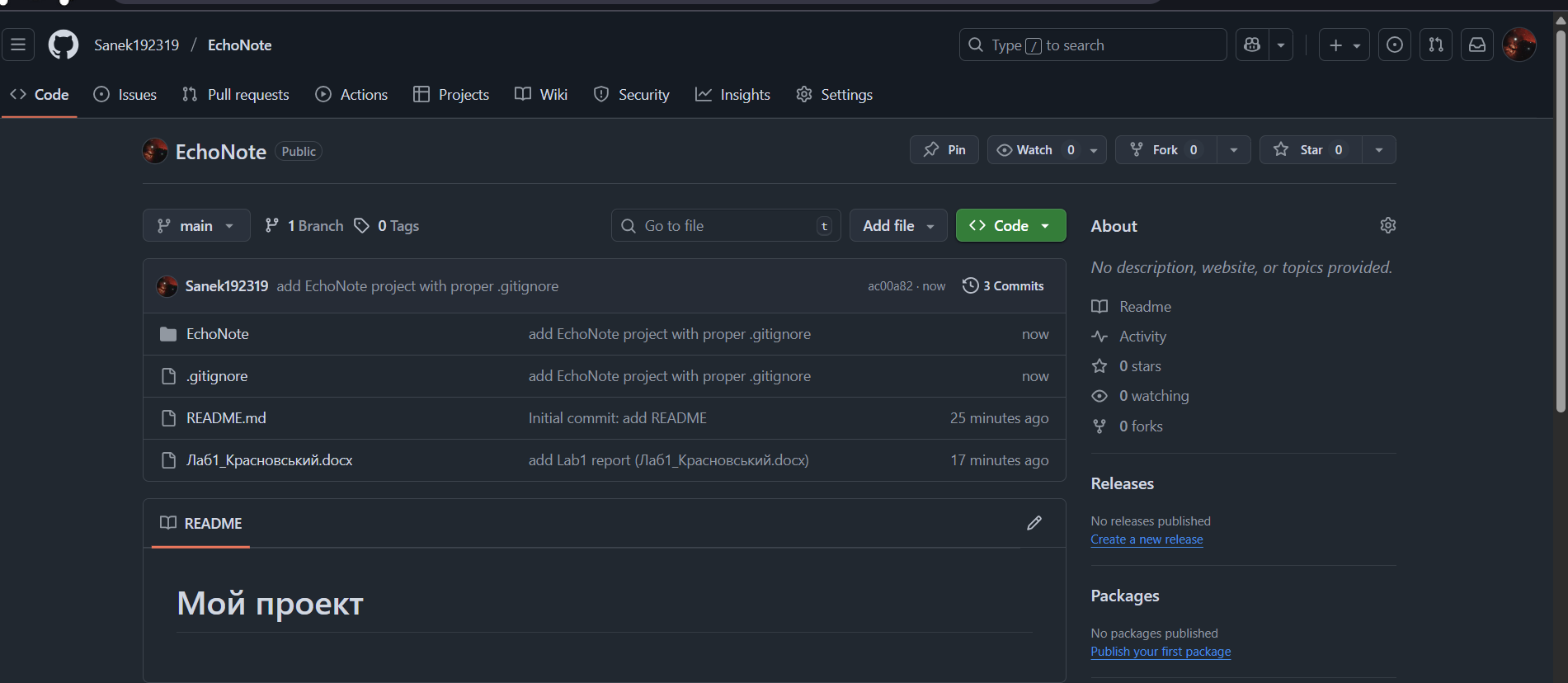


Рисунок 2.6 – Проєкт завантажений на GitHub

**Посилання на репозиторій:** <https://github.com/Sanek192319/EchoNote>

**Висновок**

У ході виконання лабораторної роботи №2 було створено повноцінний шаблон настільного застосунку **EchoNote** на базі **C#/.NET 8** із використанням технології **WPF**. У середовищі **Visual Studio 2022** проведено початкове налаштування проєкту, вибрано цільову платформу .NET 8.0 (LTS), підключено необхідні бібліотеки через **NuGet** (Microsoft.ML.OnnxRuntime, NAudio, EntityFrameworkCore з провайдером SQLite, CommunityToolkit.Mvvm, Serilog).

Також ініціалізовано **Git-репозиторій**, додано файл .gitignore для виключення службових директорій і проведено перший коміт із завантаженням на GitHub. Отримано практичні навички організації структури проєкту, керування залежностями, роботи з системою контролю версій та розгортання репозиторію у віддаленому сховищі.

Результатом роботи є готовий до подальшої розробки каркас застосунку EchoNote, що слугуватиме надійною основою для реалізації функціоналу офлайн-розпізнавання мовлення, збереження транскриптів та семантичного пошуку.