

УДК 004.7

*Кунун Т.В., к.т.н., ст. викладач,  
Сидоров ЮМ., здобувач,  
Національний університет «Одеська політехніка»*

## **ADMINISTRATOR UTILITY: УТИЛІТА ЦЕНТРАЛІЗОВАНОГО КЕРУВАННЯ СИСТЕМНИМИ ОБМЕЖЕННЯМИ WINDOWS**

В роботі розглянута створення прикладної утиліти для системних адміністраторів, яка дозволяє централізовано керувати службами, політиками безпеки та апаратними обмеженнями операційної системи Windows з одного інтерфейсу.

Застосунок має модульну архітектуру, побудовану на розділенні відповідальностей між окремими бібліотеками (assemblies). Кожен модуль відповідає за окремий аспект роботи програми, що спрощує підтримку та розширення функціоналу.

Основні модулі застосунку

AdministratorUtility - основний виконуваний застосунок, що відповідає за ініціалізацію інтерфейсу користувача.

AdministratorUtility.Core - бізнес-логіка, пов'язана з керуванням службами, пристроями та обробкою адміністративних операцій.

AdministratorUtility.Mvvm - інфраструктура MVVM, базові ViewModel, механізми обміну повідомленнями між компонентами.

AdministratorUtility.Windows - інкапсульований доступ до можливостей Windows API.

AdministratorUtility.Storage - робота з локальною базою даних SQLite.

AdministratorUtility.FileStorage - збереження та експорт даних у файлові формати.

AdministratorUtility.Translations - підтримка багатомовного інтерфейсу.

AdministratorUtility.Shared - спільні структури даних та допоміжні утиліти.

Такий розподіл дозволяє ізолювати системну логіку від інтерфейсу користувача та забезпечує чисту архітектуру застосунку.

Для реалізації адміністративного функціоналу застосунок використовує можливості Windows API. Доступ до системних функцій здійснюється через спеціалізований модуль, що дозволяє інкапсульовати виклики API та забезпечити безпечну і контрольовану взаємодію з операційною системою. Застосунок використовує Windows API для: керування станом та типом запуску служб Windows; отримання інформації про апаратні пристрої та їхні класи; увімкнення та вимкнення пристроїв.

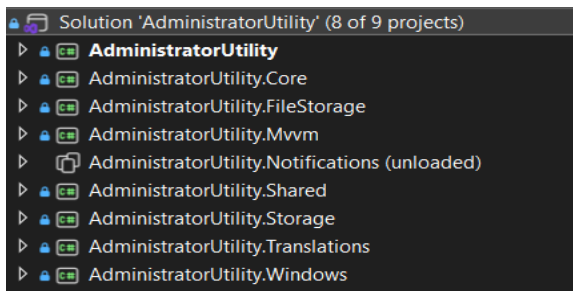


Рисунок 1 – Структура проекту

Такий підхід дозволяє реалізувати необхідний функціонал без прямого втручання користувача в системні інструменти.

Графічний інтерфейс застосунку реалізований з урахуванням зручності та швидкості роботи системного адміністратора. Інтерфейс поділений на вкладки відповідно до типів керованих об'єктів.

Основні вкладки застосунку:

Favorites - швидкий доступ до обраних служб і пристроїв;

Services - керування службами Windows;

Devices - керування пристроями та класами пристроїв;

Logs - перегляд журналу адміністративних дій;

Окремо передбачено вікно налаштувань застосунку (Settings), яке дозволяє адаптувати інтерфейс під потреби користувача.

Користувач може обрати мову інтерфейсу (українська або англійська), а також тему оформлення - системну, світлу або темну.

Застосунок надає можливість перегляду списку системних служб Windows та виконання над ними основних адміністративних дій. Користувач може змінювати стан служб, тип їх запуску або видалити з РС, відкривати розташування служб в реєстрі та файловій системі, а також додавати служби до списку обраних. Інформація про служби може бути збережена у файли різних форматів для подальшого аналізу або звітності.

Застосунок дозволяє переглядати класи пристроїв та пов'язані з ними апаратні компоненти. Передбачено можливість увімкнення та вимкнення пристроїв, перегляду інформації в системному реєстрі, додавання пристроїв до списку обраних та експорту даних у файл.

### Список використаних джерел:

1. Griffiths I. Programming C# 10: Build Cloud, Web, and Desktop Applications. Sebastopol : O'Reilly Media, 2022. P. 810.
2. Martin R. C. Clean Architecture: A Craftsman's Guide to Software Structure and Design. Boston : Prentice Hall, 2017. P. 432.
3. Microsoft. .NET documentation [Електронний ресурс]. Режим доступу: <https://learn.microsoft.com/dotnet>