Єзерський Кирил Ігоревич

Національний університет «Одеська політехніка», Україна

Кунуп Тетята Василівна, канд. техн. наук

Національний університет «Одеська політехніка» , Україна

Рудніченко Микола Дмитрович, канд. техн. наук, доцент

Національний університет «Одеська політехніка» , Україна

**ПРОЕКТ СТРУКТУРИ КЛАСІВ ТА ВИКЛИКІВ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ УПРАВЛІННЯ ДАНИМИ ФАЙЛОВОЇ СИСТЕМИ**

Вступ. Файловий менеджер (file manager) – це комп'ютерна програма, що надає інтерфейс користувача для роботи з файловою системою і файлами. Файловий менеджер дозволяє виконувати найбільш часті операції над файлами – створення, відкриття/програвання/перегляд, редагування, переміщення, перейменування, копіювання, видалення, зміна атрибутів і властивостей, пошук файлів і призначення прав [1]. Крім основних функцій, багато файлових менеджерів включають ряд додаткових можливостей, наприклад, такі, як робота з мережею (через FTP, NFS і т. п.), резервне копіювання, управління принтерами і пр. Часто вони супроводжуються доповнюваними утилітами для поліпшення життя користувача [2]. Для багатьох користувачів улюблений файловий менеджер часто виступає в роликах оболонки, змінюючи частину стандартних засобів роботи з файлами, наявними в операційній системі.

На розробленій діаграмі класів наведено склад наступних класів та інтерфейсів: MainWindows, клас для забезпечення роботи головної форми користувальницького інтерфейсу розробленого проекту програмного забезпечення. GeneralConfigSystem, клас для зберігання та опрацювання конфігураційних налаштувань роботи програмного забезпечення. Resources, клас, який є необхідним для обробки процедур та зберігання шляхів до зовнішніх файлів, які використано у проекті. Settings, клас, що здійснює збереження структур даних по користувальницьких налаштуваннях. HelpSystem, клас довідки. SessionFlags, клас, що зберігає значення флагів обирання відповідних опцій у інтерфейсі користувача. IHelpSystem, інтерфейс, що описує основні методи роботи класу довідки. IGeneralConfigSystem, інтерфейс, що описує методи та функції виконання процедур конфігурації.

Фрагмент діаграми класів проекту наведено на рис.1.

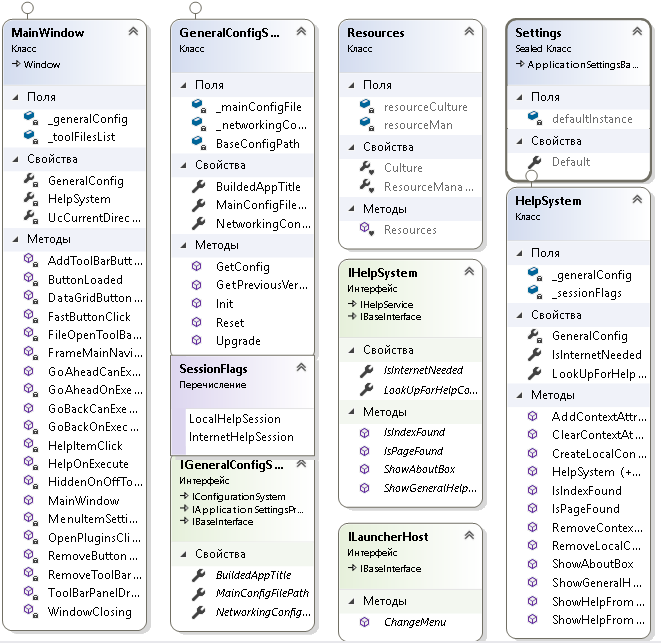


Рисунок 1 – Фрагмент діаграми класів проекту програмного забезпечення

Для чіткості виконання процесу завдання взаємозв’язків між розробленими сутностями проекту необхідно надати схему викликів та використання класів у пакетах програмного рішення. Схематичний вигляд послідовності викликів пакета головного модуля програмного забезпечення наведено на рис.2. Рішення містить 5 пакетів: UltimaCmd (складається з класу App та класу MaiinWindow); UltimaCmd.Configuration (складається з класу GeneralConfigSystem і інтерфейсу IGeneralConfigSystem); UltimaCmd.Properties (містить класи налаштувань та ресурсів – Settings та Resources); UltimaCmd.Help (складається з класу HelpSystem, інтерфейсу IHelpSystem та структури SessionFlags); UltimaCmd.InternailLogic (містить інтерфейс ILauncherHost).

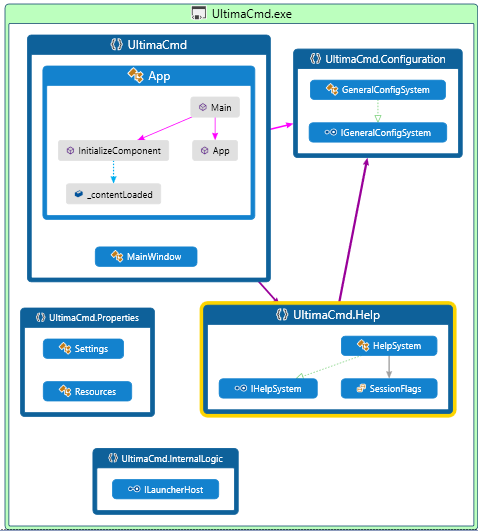


Рисунок 2 – Схематичний вигляд послідовності викликів пакета головного модуля програмного забезпечення

Висновки**.** Розроблений проект структури класів та викликів програмного забезпечення управління даними файлової системи є основою для наповнення його функціоналом у середовищі розробки та створенні інтерфейсу користувача.

**Література**

1. Panek C. Windows Operating System Fundamentals. - Sybex; John Wiley & Sons, Inc., 2020. -367 p.
2. Silberschatz A., Galvin P.B. Operating System Concepts. - Wiley, 2013. - 782 p..