

DOI: <https://doi.org/10.36910/6775-2524-0560-2024-55-28>

УДК 004.75

Христинець Наталія Анатоліївна, к.т.н., доцент

<https://orcid.org/0000-0002-4836-7632>

Вікторова Яна Андріївна, студентка

Жуковська Софія Юрїївна, студентка

Радіон Юлія Андріївна, студентка

Луцький національний технічний університет, м. Луцьк, Україна

РОЗРОБКА ФАЙЛОВОГО МЕНЕДЖЕРА ОПЕРАЦІЙНОЇ СИСТЕМИ

Христинець Н.А., Вікторова Я.А., Жуковська С.Ю., Радіон Ю.А. Розробка файлового менеджера ОС. В цій роботі описується розробка та реалізація простого файлового менеджера для операційної системи. Файловий менеджер - це програма, яка надає користувачеві інтерфейс для взаємодії з файловою системою комп'ютера. Він дозволяє створювати, видаляти, копіювати, переміщати та шукати файли та папки, а також переглядати їх властивості. Файлові менеджери є невід'ємною частиною будь-якої операційної системи. Функціонал цього файлового менеджера включає можливість працювати з різними типами файлів та забезпечує зручний спосіб їх упорядкування. Інтерфейс програми був розроблений таким чином, щоб бути інтуїтивно зрозумілим для користувачів з різним рівнем технічної підготовки. Важливою особливістю є підтримка операцій з великим обсягом даних, що підвищує ефективність роботи користувачів. У процесі розробки було враховано вимоги безпеки для захисту даних від несанкціонованого доступу. Крім того, файловий менеджер підтримує інтеграцію з іншими додатками, що розширює його функціональні можливості.

Ключові слова: файловий менеджер, файлова система, операції над файлами, Python, командний рядок.

Khrystynets N., Viktorova Y., Zhukovska S., Radion Yu. OS file manager development. This paper describes the development and implementation of a simple file manager for the operating system. A file manager is a program that provides the user with an interface to interact with the computer's file system. It allows you to create, delete, copy, move, and search for files and folders, as well as view their properties. File managers are an essential part of any operating system. The function of the file manager is to enable the reader to follow the file's instructions and thus avoid a serious injury. Interpreters are proving to be a great help to Chinese, but they are incredibly dangerous for the brains of young people. The important thing is to bring about a reduction in the number of animals, which in turn leads to the effective breeding of robots. In the process, the pink ball turned out to be a nightmare for the security forces of the unsanctioned government. Indeed, Filovy Medvedev is attempting to integrate into other dimensions, he is expanding yoga's functional capabilities.

Keywords: file manager, file system, file operations, Python, command line.

Постановка наукової проблеми. У наш час практично неможливо уявити будинок людини без комп'ютера. Ви не знайдете компанії, офісу, фабрики чи банку, які не надають такі пристрої. Люди використовують персональні комп'ютери для зберігання необхідної інформації, даних і файлів. Але якими б корисними не були ПК, технологія з кожним днем стає все більш необхідною та популярною, щоб уникнути значного дискомфорту, спричиненого використанням комп'ютера. Великі системні блоки, потреба у великому обсязі пам'яті та спеціальному робочому місці, значна кількість шнурів, випадки поломок - це проблеми, які необхідно вирішувати якнайшвидше для продовження технічного прогресу людства. В сучасному цифровому світі, де ми постійно генеруємо та зберігаємо великі обсяги даних, ефективне керування файлами стає життєво необхідною потребою. Саме тут на допомогу приходять файлові менеджери – програмні інструменти, які надають користувачам зручний інтерфейс для взаємодії з файловою системою комп'ютера.

Аналіз досліджень. Існує багато різних файлових менеджерів, доступних для різних операційних систем. Деякі популярні файлові менеджери для Windows включають Провідник Windows, Total Commander та FreeCommander. Деякі популярні файлові менеджери для Linux включають Nautilus, Dolphin та Krusader. Файлові менеджери відрізняються за функціональністю, можливостями та інтерфейсом користувача.

Мета роботи: Метою цієї роботи є розробка простого файлового менеджера для операційної системи, який буде підтримувати основні операції над файлами та папками. Файловий менеджер написаний засобами мови програмування Python.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження. В сучасному цифровому світі, де ми постійно генеруємо та зберігаємо великі обсяги даних, ефективне керування файлами стає життєво необхідною потребою. Саме тут на допомогу приходять файлові менеджери – програмні інструменти, які надають користувачам зручний інтерфейс для взаємодії з файловою системою комп'ютера. Файлові менеджери надають графічний інтерфейс користувача (GUI), який полегшує навігацію по файловій системі та виконання операцій над файлами та папками. Існує багато різних файлових менеджерів, доступних для різних операційних систем,

кожен з власним набором функцій та можливостей. Файлові менеджери є невід'ємним інструментом для керування файлами та папками на комп'ютері. Вони надають графічний інтерфейс користувача, який полегшує навігацію по файловій системі та виконання операцій над файлами та папками. Файлові менеджери можуть використовуватися для створення, видалення, копіювання, переміщення та пошуку файлів та папок, а також для перегляду їх властивостей. Вибір відповідного файлового менеджера залежить від індивідуальних потреб та вподобань користувача. Деякі користувачі можуть віддавати перевагу файловим менеджерам з багатим набором функцій, тоді як інші можуть віддавати перевагу простішим та зручнішим у використанні.

Менеджер файлів — це комп'ютерна програма, яка забезпечує інтерфейс користувача, необхідний для роботи з каталогами та файлами. Менеджер файлів дозволяє виконувати основні операції з файлами - створювати, відкривати і перемішувати, переглядати, редагувати, видаляти, перейменовувати, шукати файли, копіювати, змінювати властивості і властивості, призначати дозволи. Менеджери файлів забезпечують більш зручний та інтуїтивно зрозумілий спосіб спілкування з вашим ПК, ніж операційна система (ОС). Однією з найвідоміших ранніх програмних оболонок був Norton Commander. Його розробив американський програміст Пітер Нортон. Цей файловий менеджер чітко відображає на моніторі всю файлову структуру пристрою: диски, каталоги та файли. Використання такої програми позбавляє від необхідності вводити складні команди MS-DOS у командному рядку.

В даний час в операційній системі Windows є свій інструмент для візуальної роботи з файловою системою - вбудована програма Провідник. Однак, незважаючи на це, файлові менеджери все ще користуються великою популярністю. Таким чином, файловий менеджер - це програма, спеціально розроблена для полегшення спілкування користувача з командами операційної системи. Ці програми працюють під керуванням операційної системи і займають проміжне положення між операційною системою та додатками. Крім основних функцій, багато додаткових функцій: мережеві операції, можливість налаштування інтерфейсу, можливість створення макросів і інших зовнішніх функцій пристроїв. Розрізняють основні типи файлових менеджерів: навігаційний і пробіл - іноді можуть підтримувати перемикання між цими режимами; дві панелі - зазвичай оснащені двома однаковими панелями для списку файлів і дерева каталогів. Представником навігаційного (космічного) файлового менеджера є звичайний Windows Explorer. Це не найкраща, але корисна програма. Він не завжди здатний виконувати операції з великими файлами, наприклад, коли вам потрібно перемістити або перейменувати сотню фотографій. При роботі з великою кількістю файлів Провідник може значно уповільнити роботу системи. Але найсуттєвішим недоліком є відсутність другої панелі для копіювання або переміщення файлів. Інші відомі навігаційні (просторові) файлові менеджери включають: Pocket Total Commander, Directory Opus Magellan, Software Standard File Manager, Resco Explorer, Workbench — включено в AmigaOS, Konqueror — включено в KDE, Dolphin — включено в KDE 4, Nautilus (файловий менеджер) – поставляється з GNOME, Thunar – поставляється з Xfce, Bynarys Smart Explorer, ROX-Filer – поставляється з ROX Desktop. У другому типі менеджера (двопанельному), який використовується для багатьох операцій, вікно програми розділене на дві частини. При цьому робота з файлами стає зручнішою та швидшою. А якщо врахувати, що більшість файлових менеджерів дозволяють контролювати всі дії з клавіатури, то швидкість роботи зростає в кілька разів. Найвідоміші ортодоксальні (двопанельні) файлові менеджери: Norton Commander, Total Commander, DOS Navigator, Volkov Commander, PIE Commander, Microsoft Windows, FreeCommander, Nomad.NET, Unreal Commander, File Navigator, GNOME Commander.

Програма Windows Explorer — це спеціальний плагін типу файлового менеджера, вбудований в операційну систему Windows. Програма призначена для навігації по файловій структурі Windows і її обслуговування (копіювання, перейменування, видалення і переміщення папок і файлів). Запуск програми здійснюється через команду Пуск/Програми/Провідник. Провідник Windows одночасно відображає вкладену структуру папок на комп'ютері (її ієрархію) і відображає внутрішню частину вибраної папки. Це дуже зручно під час копіювання та переміщення даних: просто відкрийте папку з потрібними файлами та перетягніть їх у папку призначення. Коли вам потрібно відкрити, скопіювати, перемістити, видалити, перейменувати чи іншим чином змінити папки чи файли, ви можете використовувати Провідник файлів або працювати з папками та файлами безпосередньо на робочому столі. Основною перевагою, на відміну від звичайних вікон папок на Робочому столі, є те, що програма Провідник надає можливість одночасно працювати як з вмістом правої панелі вікна, так і з усією архітектурою файлової системи комп'ютера користувача, тобто з

лівою панеллю. Це, наприклад, значно спрощує копіювання файлів із правої панелі в каталог або логічний диск, який зараз відображається на лівій панелі. При цьому немає необхідності відкривати велику кількість вікон, як це довелося б робити на Робочому столі. Робота з утилітою Провідник мало чим відрізняється від роботи зі звичайними вікнами Windows, за винятком того, що користувач може одночасно переглядати ієрархічну структуру своєї файлової системи. Він може клацнути правою кнопкою миші будь-яку папку та переглянути її вміст або клацнути будь-яку піктограму, щоб перетягнути її в інше місце.

Total Commander, колишній Windows Commander, є найпопулярнішим файловим менеджером для ОС Windows, крім Windows Explorer, і розроблений для зручної роботи з файлами та каталогами користувача. Подібно до Провідника, він дозволяє виконувати всі основні операції з файлами та каталогами, такі як копіювання, переміщення, видалення та зміни. Однак Total Commander використовує інший, більш зручний підхід до організації інтерфейсу: він має дві постійні панелі, які спрощують навігацію користувача та полегшують пошук необхідного місця для копіювання файлу. В усьому цьому допомагають постійно видимі каталог джерела файлу та каталог приймача для зручності користувача. Frigate - це дуже потужний файловий менеджер, який дозволяє швидко, інтуїтивно зрозуміло і комфортно працювати з комп'ютером. На сьогоднішній день Frigate є найбільш функціональним та потужним засобом для роботи з файлами.

Основні характеристики Frigate: Цей файловий менеджер включає в себе підтримку багатозадачності при виконанні файлових операцій, що дозволяє користувачу виконувати декілька операцій одночасно. Він також забезпечує підтримку усіх відомих архівів, що полегшує роботу з архівами. Навігація в менеджері є зручною завдяки "швидким" папкам та історією переміщення по папках. Така система навігації полегшує пошук та доступ до потрібних файлів. Крім цього, у менеджері доступні корисні утиліти, такі як калькулятор, системна інформація, телефонна книга, зберігач паролів та інші, що розширює його функціональність і корисність для користувача. Також, вбудований перегляд (F3) дозволяє швидко переглядати RTF-текст і файли DBF, а вбудований редактор (F4) надає можливість редагувати текст з підсвічуванням синтаксису і RTF-текстові файли. Менеджер також забезпечує відновлення файлів після обриву копіювання, що є важливою функцією для збереження даних. Крім цього, він підтримує перегляд і редагування MP3-тегів, вдосконалений пошук та файловий фільтр для швидкого відбору типів файлів. Користувач також може працювати з картинками, оскільки менеджер підтримує всі популярні графічні формати. Інші можливості Frigate включають широку функціональність, гнучкість у налаштуваннях та інтеграцію з іншими додатками.

У нашій роботі ми використали мову програмування Python, оскільки основними архітектурними особливостями її є динамічна типізація, механізм обробки винятків, автоматичне керування пам'яттю, підтримка багатопоточних обчислень і повний самоаналіз. Динамічна семантика та прив'язки, а також високорівневі структури даних роблять цю мову привабливою для прискореної розробки додатків. Python підтримує велику кількість парадигм програмування, включаючи об'єктно-орієнтовану, процедурну, імперативну, функціональну та аспектно-орієнтовану. Крім того, Python є кросплатформною і не вимагає додаткових зусиль для перенесення з однієї платформи на іншу.

Інтерпретатор написаний на C, а вихідний код доступний для будь-яких маніпуляцій. Якщо необхідно, можна вставити його у свою програму та використовувати, як вбудовану оболонку. Або, написавши свої доповнення на C, отримати «просунутий» інтерпретатор з новими можливостями. Корисним є функціонал розширення для програм, які потребують подальшого налагодження. Підтримка модулів та пакетів модулів значно збільшує повторне використання коду, що корисно під час розробки великих програм. Інтерпретатор Python і стандартна бібліотека доступні у скомпільованій та вихідній формі на всіх основних платформах. Для виконання програми потрібно використання бібліотеки send2trash. Вона використовується для перенесення файлів до смітника. Її можна встановити за командою `pip install send2trash`. Обговоримо основні функції, які ми використали у цій роботі.

Імпорт необхідних модулів:

```

1  #! python3
2
3  import sys
4  import os
5  import shutil
6  import send2trash

```

Рис. 1. Початок програми, підключення бібліотек

sys: Цей модуль використовується для доступу до вбудованих функцій та змінних Python, таких як exit(). os: Цей модуль використовується для взаємодії з операційною системою, наприклад, для отримання доступу до файлів та каталогів, переміщення каталогів та видалення файлів. shutil: Цей модуль використовується для копіювання, переміщення та видалення файлів та каталогів. Send2trash: Цей модуль використовується для надсилання файлів та каталогів до кошика.

Змінна drives для зберігання списку доступних дисків:

```

10 # Зберігає кожний підключений до ПК диск у списку.
11 drives = [chr(x) + ':' for x in range(65, 90) if os.path.exists(chr(x) + ':')]

```

Рис. 2. Функція збереження списку доступних дисків

Цей код використовує цикл for, щоб перебрати символи з A по Z (65 - 90 в ASCII). Для кожного символу перевіряється, чи існує диск з такою літерою, використовуючи os.path.exists(). Якщо диск існує, його додається до списку drives.

Функція listDirectories() для виведення списку файлів та каталогів:

```

13 # Виводить список кожної теки та файлу, що знаходяться в поточній робочій директорії.
14 10 usages
15 def listDirectories():
16     listdir = os.listdir(os.getcwd())
17     for x in listdir:
18         print(x)

```

Рис. 3. Функція для виведення списку файлів та каталогів

Ця функція використовується для виведення списку файлів та каталогів у поточній робочій директорії. Вона використовує os.listdir() для отримання списку елементів у поточній директорії. Потім вона циклічно перебирає цей список і друкує назву кожного елемента.

Головний цикл while True для роботи менеджера файлів:

```

19 while True:
20     print("1. Відкрити файл/теку \n2. Переіменувати \n3. Перемістити та вставити \n4. Скопіювати та вставити \n5. Видалити\n")
21     result = input("Оберіть одну з наступних опцій: ")
22
23     if result == '1':
24         # Головний екран
25         print("\nВидки доступу:\n1. Документи\n2. Відео\n3. Зображення\n4. Завантаження\n")
26
27         print('Диски: ')
28         for x in range(len(drives)):
29             print(str(5 + x) + '. ' + drives[x])
30
31         while True:
32             inp = input("\nВведіть ваш вибір: ")
33
34             if inp == '1':
35                 path = 'C:\\Users\\$USERNAME\\Documents'
36                 os.chdir(os.path.expandvars(path))
37                 break
38
39             elif inp == '2':
40                 path = 'C:\\Users\\$USERNAME\\Videos'
41                 os.chdir(os.path.expandvars(path))
42                 break
43
44             elif inp == '3':
45                 path = 'C:\\Users\\$USERNAME\\Pictures'
46                 os.chdir(os.path.expandvars(path))

```

Рис. 4. Головний цикл while True. Частина 1

Цикл while True забезпечує безперервну роботу менеджера файлів, поки користувач не вирішить вийти. У середині циклу: Відображається меню з 5 опціями. Користувач вводить свій вибір. Вибір обробляється за допомогою вкладених операторів if elif else. Якщо введено неправильний вибір, друкується повідомлення про помилку.

Обробка вибору користувача відбувається через відкриття файлу/папки. Фрагмент коду нижче використовується для надання швидкого доступу до документів, відео, зображень та завантажень. Користувач може вибрати один із цих місць або перейти на диск.

```

44         elif inp == '3':
45             path = 'C:\\Users\\$USERNAME\\Pictures'
46             os.chdir(os.path.expandvars(path))
47             break
48
49         elif inp == '4':
50             path = 'C:\\Users\\$USERNAME\\Downloads'
51             os.chdir(os.path.expandvars(path))
52             break
53
54         elif inp in drives:
55             os.chdir(inp + '\\')
56             break
57
58         else:
59             print('Помилка\nВведіть правильний ввід або назву диска.\n')
60

```

Рис. 5. Головний цикл while True. Частина 2

Після вибору диска або місця код переходить до каталогу та відображає список файлів та каталогів у ньому. Користувач може вибрати файл, щоб відкрити його, або каталог, щоб перейти до нього. Користувач також може ввести exitManager, щоб вийти з менеджера файлів, або backManager, щоб перейти на один каталог вище. **Перейменувати:** Користувач може вибрати диск, щоб перейти до нього. Після вибору диска код відображає список файлів та каталогів у ньому. Користувач може вибрати файл для перейменування. Користувач буде запитуватися на введення нового імені, а потім старе ім'я буде змінено на нове за допомогою функцій os.rename. Користувач також може ввести exitManager, щоб вийти з менеджера файлів, або backManager, щоб перейти на один каталог вище.

```

85         if result == '2':
86             print("Ви вибрали перейменувати")
87             print('Диски: ')
88             for x in range(len(drives)):
89                 print(str(1 + x) + '. ' + drives[x])
90
91             while True:
92                 inp = input("\nВведіть ваш вибір: ")
93
94                 if inp in drives:
95                     os.chdir(inp + '\\')
96                     break
97                 else:
98                     print('Помилка\nВведіть правильну назву диска.\n')
99

```

Рис. 6. Функція для виконання опції «Перейменувати»

Перемістити та вставити: Цей код реалізує файловий менеджер з можливістю перегляду та переміщення файлів та каталогів. Після запуску користувачу відображається список дисків. Він може обрати диск, щоб перейти до нього. Після вибору диска відображається список файлів та каталогів у ньому. Користувач може обрати файл або каталог для переміщення. Потім користувачу буде запропоновано ввести місце призначення, і обраний файл або каталог буде переміщено до нового розташування за допомогою функцій shutil.move(). Крім того, користувач може ввести команду exitManager, щоб вийти з менеджера файлів, або backManager, щоб перейти на один каталог вище.

Скопіювати та вставити: Цей код створює файловий менеджер з можливістю перегляду та копіювання файлів та каталогів. При запуску програми користувачу відображається список дисків, з якими він може працювати. Він може обрати диск для подальшої роботи. Після вибору диска відображається список файлів та каталогів, що містяться на цьому диску. Користувач може обрати файл або каталог, який він бажає скопіювати. Після цього він буде запитаний про нове місце розташування, куди слід скопіювати вибраний файл або каталог. Копіювання виконується за допомогою функцій shutil.copytree() для каталогів та shutil.copy2() для файлів. Крім того, користувач може ввести команду exitManager, щоб вийти з менеджера файлів, або backManager, щоб перейти на один каталог вище.

Видалити: Цей код включає опцію для користувача вибрати, чи він хоче видалити файл або каталог назавжди, чи перемістити його до кошика. Якщо користувач обере видалення назавжди, він отримає попередження про можливі наслідки цієї дії та буде запрошений підтвердити видалення. Якщо видалення підтверджено, відповідний файл або каталог буде видалено назавжди за допомогою функцій `os.remove()` для файлів та `shutil.rmtree()` для каталогів.

У випадку, якщо користувач вирішить перемістити файл або каталог до кошика, вони будуть переміщені до кошика за допомогою модуля `send2trash`. Це надає можливість відновлення файлів або каталогів у разі необхідності, адже вони не видаляються остаточно, а лише переміщуються до системного кошика.

Обробка помилок вводу користувача в файловому менеджері включає перевірку правильності введених користувачем опцій. Коли користувач вводить опцію, код перевіряє, чи вона є дійсною та чи належить до діапазону доступних опцій. Це реалізується за допомогою операторів `if`, `elif` і `else`. Якщо введена опція неправильна, код виводить повідомлення про помилку та просить користувача ввести опцію знову, поки не буде введено правильну.

Також в файловому менеджері передбачена обробка помилок вводу назви диска, якщо користувач виконує дії, такі як перейменування, переміщення, копіювання або видалення файлів або каталогів. Код перевіряє правильність введеної назви диска, наприклад, чи існує такий диск та чи є введена назва диском. Якщо введена назва диска неправильна, виводиться повідомлення про помилку, і користувачу пропонується ввести правильну назву.

Додаткові функції, такі як `backManager` та `exitManager`, реалізовані для полегшення навігації користувача в менеджері файлів. Функція `backManager` дозволяє користувачеві перейти на один рівень вище в поточній робочій директорії, щоб швидше навігуватися по системі. Функція `exitManager` дає можливість користувачеві вийти з менеджера файлів у будь-який момент. Обидві ці функції спрощують використання менеджера файлів та зроблюють його більш зручним для користувача.

Вихід: Користувач може вийти з менеджера файлів, ввівши `exit` або `exitManager` у головному меню. У виводі цієї програми можна отримати такі результати:

```
C:\Users\Asus\PycharmProjects\pythonProject1\.venv\Scripts\python.exe C:\Users\Asus\PycharmProjects\pythonProject1\main.py
Ласкаво просимо до файлового менеджера на Python!

1. Відкрити файли/теки
2. Перейменувати
3. Перемістити та вставити
4. Скопіювати та вставити
5. Видалити

Оберіть одну з наступних опцій:
```

Рис. 7. Початок виводу програми

Наприклад, якщо ви оберете опцію 1, вивід буде таким:

```
Оберіть одну з наступних опцій: 1

Швидкий доступ:
1. Документи
2. Відео
3. Зображення
4. Завантаження

Диски:
5. C:
6. D:
7. E:

Введіть ваш вибір:
```

Рис. 8. Вивід опції 1


```
Введіть ваш вибір: 1
cc_20220319_000048.reg
desktop.ini
IISExpress
iZotope
LabVIEW Data
MATLAB
My Music
My Pictures
My Videos
My Web Sites
National Instruments
ViberDownloads
Visual Studio 2022
Zoom
Без назви-1.dxf
Без назви-1_1.png
Без назви-1_2.png
Без назви-1_3.png
Настраиваемые шаблоны Office
Настроенные шаблоны Office
Пользовательские шаблоны Office
```

Рис. 9. Вивід тек для опції 1 на диску 1. Частина 1

```
Напишіть "exitManager", щоб вийти з менеджера файлів.
Напишіть "backManager", щоб перейти вище на один каталог.

Оберіть файл/теку:
```

Рис. 10. Вивід тек для опції 1 на диску 1. Частина 2

Якщо обрати іншу опцію зі списку, вивід буде побудований по тому ж принципу. Першою дією буде вибір опції, наступним кроком потрібно буде обрати диск з яким постає необхідність взаємодіяти. Далі обираються папки для роботи.

Висновки.

У сучасному світі, де комп'ютери та цифрові технології стали невід'ємною частиною нашого життя, виникає гостра потреба у покращенні та спрощенні роботи з ними. Існує ряд проблем, з якими стикаються користувачі персональних комп'ютерів. Для вирішення цих проблем необхідні інноваційні рішення, спрямовані на покращення ергономіки, простоти використання та надійності комп'ютерних систем. Розробка простого файлового менеджера є одним із актуальних кроків у цьому напрямку. Отож, був розроблений простий файловий менеджер для операційної системи, який відповідає поставленим цілям. Файловий менеджер підтримує основні операції над файлами та папками та написаний чітким та зрозумілим кодом.

Список бібліографічного опису

1. DSpace. ELAKPI: Репозитарій КІІ ім. Ігоря Сікорського. URL: <https://ela.kpi.ua/server/api/core/bitstreams/98d6a1b6-b74a-4c04-9e26-6ce460015be6/content> (дата звернення: 10.05.2024).
2. Міністерство освіти і науки України. Інститут інформатики та радіоелектроніки, Факультет комп'ютерних наук і технологій. Кафедра програмних засобів. РОЗРОБКА ПРОГРАМИ ФАЙЛОВОГО МЕНЕДЖЕРА ДЛЯ ОПЕРАЦІЙНОЇ СИСТЕМИ ANDROID. FILE MANAGER DEVELOPMENT FOR ANDROID OS. URL: https://eir.zntu.edu.ua/bitstream/123456789/10723/1/BR_Burdakova.pdf (дата звернення: 10.05.2024).

References

- 1.DSpace. ELAKPI: KPI Repository. Igor Sikorsky. URL: <https://ela.kpi.ua/server/api/core/bitstreams/98d6a1b6-b74a-4c04-9e26-6ce460015be6/content> (Last accessed: 10.05.2024).
- 2.Ministry of Education and Science of Ukraine. Institute of Informatics and Radioelectronics, Faculty of Computer Science and Technology. Department of Software. DEVELOPMENT OF A FILE MANAGER APPLICATION FOR THE ANDROID OPERATING SYSTEM. FILE MANAGER DEVELOPMENT FOR ANDROID OS. URL: https://eir.zntu.edu.ua/bitstream/123456789/10723/1/BR_Burdakova.pdf (Last accessed: 10.05.2024).