

Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна
Навчально-науковий інститут комп'ютерних наук та штучного інтелекту
Кафедра комп'ютерних систем та робототехніки

«Затверджую»
в.о. завідуючого кафедри
комп'ютерних систем та робототехніки
_____ к. ф.-м. н., доцент Максим Хруслов
«___» червня 2025 р.

Пояснювальна записка


до кваліфікаційної роботи
бакалавра


на тему: **«КОМП'ЮТЕРНА МОДЕЛЬ СИСТЕМИ АНАЛІЗУ ТА
УПРАВЛІННЯ МЕТАДАНИМИ НАУКОВИХ ПУБЛІКАЦІЙ»**


Спеціальність 151 – Автоматизація та комп'ютерно-інтегровані технології
Галузь знань 15 – Автоматизація та приладобудування
Освітня програма «Автоматизація та комп'ютерно-інтегровані технології»

Захищено на засіданні
Екзаменаційної комісії № 46
протокол № __ від __.06.2025 р.
Оцінка _____ / _____

Голова Екзаменаційної комісії
_____ ЧУГАЙ А.М.

Виконала:
Студентка групи КУ– 41
БОРОВКОВА Віра Андріївна 

Керівник: в.о. завідуючого кафедри
комп'ютерних систем та
робототехніки, к. ф.-м. н., доцент
ХРУСЛОВ Максим Михайлович 

Рецензент: доцент кафедри
інтелектуальних систем і технологій
ХНУ ім. В. Н. Каразіна, к. ф.-м. н.
КАРАСЬ Ірина Вячеславівна 

АНОТАЦІЯ

Пояснювальна записка до кваліфікаційної роботи бакалавра складається зі вступу, чотирьох розділів, висновків, списку використаних джерел і чотирьох додатків. Загальний обсяг роботи складає 58 сторінок, із яких 38 сторінок основної частини з 15 рисунками, 2 таблицями, 20 найменуваннями списку використаних джерел та чотирма додатками.

Метою кваліфікаційної роботи є розробка комп'ютерної моделі системи аналізу та управління метаданими наукових публікацій, що дозволяє автоматизувати процес додавання метаданих до PDF-файлів на основі даних з CSV-файлу.

Об'єкт дослідження – процеси управління метаданими наукових публікацій.

Предмет дослідження – методи та засоби автоматизації додавання метаданих до PDF-документів з використанням мови програмування Python.

Ключові слова: *метадані, наукові публікації, PDF, CSV, Python, обробка даних.*

ABSTRACT

The explanatory note to the bachelor's qualification work consists of an introduction, four sections, conclusions, a list of sources used, and four appendices. The total volume of the work is 58 pages, of which 38 pages are the main part with 15 figures, 2 tables, 20 names of the list of sources used, and four appendices.

The purpose of the qualification work is to develop a computer model of a system for analyzing and managing metadata of scientific publications, which allows you to automate the process of adding metadata to PDF files based on data from a CSV file.

The object of the study is the processes of managing metadata of scientific publications.

The subject of the study is methods and tools for automating the addition of metadata to PDF documents using the Python programming language.

Keywords: *metadata, scientific publications, PDF, CSV, Python, data processing.*

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	6
ВСТУП.....	7
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ УПРАВЛІННЯ МЕТАДАНИМИ.....	9
1.1 Поняття метаданих: типи, функції, застосування	9
1.2 Стандарти метаданих.....	12
1.2.1 Dublin Core.....	12
1.2.2 Resource Description Framework (RDF).....	13
1.2.3 Extensible Markup Language (XML).....	14
1.3 Формати документів та інструменти для роботи з метаданими.....	17
1.3.1 Файл PDF і його метадані.....	17
1.3.2 Формат CSV як засіб зберігання метаданих.....	19
1.3.3 Бібліотеки Python для роботи з PDF, CSV та метаданими...	21
Висновки за розділом 1.....	22
РОЗДІЛ 2. АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ.....	23
2.1 Огляд програмних засобів.....	23
2.1.1 PyPDF2.....	23
2.1.2 pdfcrow.....	24
2.1.3 PyMuPDF.....	24
2.2 Порівняльний аналіз.....	25
Висновки за розділом 2.....	27
РОЗДІЛ 3. РОЗРОБКА ПРОГРАМИ НА PYTHON.....	28
3.1 Постановка задачі.....	28
3.2 Опис алгоритму.....	28
3.3 Використані бібліотеки.....	28
3.4 Тестування програми.....	29
Висновки за розділом 3.....	36

РОЗДІЛ 4. ПІДБИТТЯ ПІДСУМКІВ І ПЕРСПЕКТИВИ РОЗВИТКУ	38
4.1 Робота програми.....	38
4.2 Можливості вдосконалення.....	38
ВИСНОВКИ.....	40
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	42
ДОДАТКИ.....	44
Додаток А.....	44
Додаток Б.....	46
Додаток В.....	48
Додаток Г.....	55

ПЕРЕЛІК СКОРОЧЕНЬ І УМОВНИХ ПОЗНАЧЕНЬ

RDF – Resource Description Framework

XML – Extensible Markup Language

PDF – Portable Document Format

XMP – Extensible Metadata Platform

CSV – Comma-Separated Values

W3C - World Wide Web Consortium

re – regular expressions

ВСТУП

У сучасному науковому середовищі спостерігається стрімке зростання обсягу електронних наукових публікацій, що потребує ефективних методів їх організації, зберігання та пошуку. Метадані, які описують ключові характеристики документів (назва, автор, ключові слова, рік публікації тощо), відіграють важливу роль у забезпеченні доступності та повторного використання наукової інформації. Проте багато наукових статей, особливо з архівів або менш відомих джерел, не містять повних або стандартизованих метаданих, що ускладнює їх індексацію та пошук.

Актуальність роботи: З метою підвищення ефективності наукової комунікації виникає потреба в автоматизованих інструментах для додавання та управління метаданими наукових публікацій. Використання мови програмування Python та її бібліотек дозволяє розробити гнучкі рішення для обробки великих обсягів даних, зокрема, для автоматичного додавання метаданих до PDF-файлів на основі інформації з CSV-файлів.

Метою кваліфікаційної роботи є розробка комп'ютерної моделі системи аналізу та управління метаданими наукових публікацій, що дозволяє автоматизувати процес додавання метаданих до PDF-файлів на основі даних з CSV-файлу.

Об'єкт дослідження – процеси управління метаданими наукових публікацій.

Предмет дослідження – методи та засоби автоматизації додавання метаданих до PDF-документів з використанням мови програмування Python

Методи дослідження: аналіз існуючих підходів до управління метаданими, розробка програмного забезпечення з використанням мови Python та бібліотек для обробки PDF і CSV, тестування та оцінка ефективності розробленої системи.

Задачі дослідження:

- 1) Проаналізувати існуючі методи та інструменти для управління метаданими наукових публікацій.
- 2) Розробити програму на Python для автоматичного додавання метаданих з CSV-файлу до PDF-документів.
- 3) Протестувати розроблену програму на реальних даних та оцінити її ефективність.

Значення роботи: Проблема, яка вирішується в кваліфікаційній роботі, полягає в необхідності ефективного та масштабованого підходу до додавання метаданих до наукових публікацій, що сприяє покращенню їх індексації. Запропоноване рішення дозволяє автоматизувати рутинні процеси, зменшити кількість помилок, прискорити підготовку публікацій до розміщення у відкритих електронних базах даних.

Область застосування: Результати роботи можуть бути впроваджені в наукових установах, університетських бібліотеках, редакціях наукових журналів, архівах та цифрових репозитаріях, де здійснюється зберігання, облік та публікація електронних наукових документів.

РОЗДІЛ 1

ТЕОРЕТИЧНІ ОСНОВИ УПРАВЛІННЯ МЕТАДАНИМИ

1.1 Поняття метаданих: типи, функції, застосування

У сучасному інформаційному середовищі метадані відіграють ключову роль у забезпеченні ефективного управління, пошуку та збереження цифрових ресурсів. Термін «метадані» походить від грецького «meta» (префікс, що означає понад, за межами) та «data» (дані) і означає «дані про дані». Іншими словами, метадані – це структурована інформація, яка описує, пояснює, локалізує або іншим чином полегшує пошук, використання та управління інформаційними ресурсами.

Метадані виконують різні функції. Вони сприяють пошуку потрібної інформації та ресурсів користувачами. Окрім цього, метадані допомагають організовувати електронні ресурси, забезпечують їхню цифрову ідентифікацію, а також сприяють архівуванню та збереженню даних. Завдяки метаданим користувачі можуть отримувати доступ до ресурсів, знаходити їх за певними критеріями, ідентифікувати, групувати схожі об'єкти, розрізняти різні ресурси та дізнаватися про їхнє розташування. Метадані, що стосуються телекомунікаційної діяльності, зокрема інтернет-трафіку, активно збираються державними органами багатьох країн. Такі дані використовуються для аналізу трафіку та можуть застосовуватися для масового спостереження.

Метадані забезпечують контекст для основних даних, дозволяючи користувачам ефективно знаходити, ідентифікувати та оцінювати інформаційні ресурси. Вони є невід'ємною частиною систем управління контентом, бібліотечних каталогів, наукових репозитаріїв та інших інформаційних систем.

Залежно від функціонального призначення, метадані поділяються на кілька основних типів:

Описові метадані (Descriptive metadata) призначені для ідентифікації та виявлення ресурсів. Вони включають такі елементи, як назва, автор, дата

створення, ключові слова, анотація тощо. Описові метадані полегшують пошук та організацію ресурсів у каталогах і базах даних.

Структурні метадані (Structural metadata) описують внутрішню організацію ресурсу, визначаючи, як різні частини даних пов'язані між собою. Наприклад, вони можуть вказувати порядок сторінок у документі, зв'язки між розділами або форматування елементів. Це особливо важливо для складних цифрових об'єктів, таких як електронні книги чи мультимедійні презентації.

Адміністративні метадані (Administrative metadata) забезпечують технічну та управлінську інформацію про ресурс. Вони можуть включати дані про формат файлу, розмір, дату створення та модифікації, права доступу, інформацію про автора або власника авторських прав. Адміністративні метадані допомагають у збереженні, управлінні та контролі доступу до ресурсів.

Довідкові метадані (Reference metadata) надають інформацію про походження, якість та відповідальність за створення метаданих. Вони включають контактні дані відповідальної особи чи організації, використані стандарти метаданих, умови доступу та використання, а також джерела, на які посилаються метадані. Довідкові метадані є важливими для оцінки достовірності та надійності метаданих, що особливо актуально при повторному використанні даних.

Статистичні метадані (Statistical metadata) - це структурована інформація, яка описує статистичні дані, процеси їх збирання, обробки та методології. Вони забезпечують контекст для розуміння, інтерпретації та ефективного використання статистичної інформації. Статистичні метадані включають визначення змінних, класифікації, схеми кодування, методи збору даних, процедури обробки та редагування даних, оцінки якості та інші відповідні деталі.

Юридичні метадані (Legal metadata) надають інформацію про правовий статус ресурсу, включаючи відомості про автора, власника авторських прав, ліцензії на використання, обмеження доступу та інші правові аспекти. Вони є

критично важливими для забезпечення дотримання прав інтелектуальної власності, управління ліцензіями та контролю доступу до ресурсів.

Важливо розуміти, що метадані є фундаментом для побудови сучасної цифрової інфраструктури. Вони забезпечують інтероперабельність між різними інформаційними системами, дозволяючи обмінюватися даними між організаціями, галузями та навіть країнами. Метадані стають основою для автоматизації процесів, машинного навчання, штучного інтелекту, а також для розвитку відкритих даних (open data) та відкритої науки (open science).

Роль метаданих у забезпеченні якості даних: Метадані відіграють ключову роль у забезпеченні якості даних. Вони допомагають відстежувати походження інформації, фіксувати зміни, що відбуваються з даними протягом життєвого циклу, та забезпечують прозорість процесів обробки. Це особливо важливо для наукових досліджень, де відтворюваність результатів залежить від точності та повноти метаданих. Якісні метадані дозволяють користувачам оцінити релевантність, достовірність та актуальність інформації.

Метадані у великих даних та штучному інтелекті: У сфері Big Data та штучного інтелекту метадані використовуються для організації, класифікації та індексації величезних масивів інформації. Вони дозволяють скорочувати витрати часу на пошук необхідних даних, автоматизувати процеси обробки та аналізу, а також забезпечують можливість побудови складних аналітичних моделей. Наприклад, у хмарних сховищах метадані допомагають управляти доступом, оптимізувати зберігання та забезпечувати відповідність вимогам безпеки.

Метадані у цифровій трансформації та електронному урядуванні: У контексті цифрової трансформації та розвитку електронного урядування метадані стають невід'ємною складовою для побудови відкритих реєстрів, електронних архівів, порталів публічної інформації. Вони забезпечують прозорість діяльності органів влади, сприяють підзвітності та підвищують довіру громадян до державних інституцій.

Зі зростанням обсягів даних та ускладненням інформаційних систем зростає й роль метаданих. Основними викликами залишаються уніфікація стандартів, забезпечення сумісності, автоматизація створення та підтримки метаданих, а також захист приватності. Перспективними напрямками є використання семантичних технологій та автоматичне генерування метаданих за допомогою штучного інтелекту.

1.2 Стандарти метаданих

У сфері управління метаданими стандарти відіграють ключову роль у забезпеченні уніфікованого підходу до опису, обміну, інтерпретації та інтеграції даних. Вони гарантують сумісність між різними системами, сприяють автоматизації процесів та підвищують якість інформаційних ресурсів. Окрім широко відомих Dublin Core, RDF та XML, існують й інші важливі стандарти та моделі, які розширюють можливості управління метаданими.

1.2.1 Dublin Core

Dublin Core Metadata Element Set являє собою набір з 15 базових елементів, призначених для опису інформаційних ресурсів. Він був розроблений у 1995 році під час конференції в Дубліні, штат Огайо, США, і з того часу став міжнародним стандартом для метаданих. Ці 15 елементів включають:

- 1) Title (Назва)
- 2) Creator (Автор)
- 3) Subject (Тема)
- 4) Description (Опис)
- 5) Publisher (Видавець)
- 6) Contributor (Співавтор)
- 7) Date (Дата)
- 8) Type (Тип ресурсу)
- 9) Format (Формат)
- 10) Identifier (Ідентифікатор)

- 11) Source (Джерело)
- 12) Language (Мова)
- 13) Relation (Зв'язок)
- 14) Coverage (Охоплення)
- 15) Rights (Права)

Dublin Core є гнучким та розширюваним, що дозволяє адаптувати його до різних галузей знань. Він широко використовується в бібліотеках, архівах, цифрових репозитаріях та веб-додатках для забезпечення інтероперабельності метаданих.

1.2.2 Resource Description Framework (RDF)

RDF – це стандарт, розроблений World Wide Web Consortium (W3C) у 2004 році для опису та обміну метаданими в Інтернеті. RDF представляє дані у вигляді триплетів виду «суб'єкт – предикат – об'єкт». Така модель дозволяє описувати ресурси та їх взаємозв'язки у формі графів, що забезпечує гнучкість, та масштабованість.

Основні принципи RDF:

1) Кожен запис у RDF складається з трьох частин: суб'єкта (ресурс, що описується), предиката (характеристика) та об'єкта (значення вказаної характеристики). Цей підхід дозволяє формувати складні мережі зв'язків між даними, які легко масштабуються та розширюються.

2) RDF-дані організовані у вигляді орієнтованого графа. У ньому вузли є ресурсами або значеннями, а ребра – відношеннями між ними, що забезпечує високу гнучкість у моделюванні інформації та дозволяє легко інтегрувати дані з різних джерел.

3) RDF підтримує кілька форматів представлення даних, зокрема RDF/XML і JSON-LD, і так полегшує обмін інформацією між різними платформами та застосунками.

До переваг RDF відносять такі властивості:

- RDF дозволяє об'єднувати інформацію з різних систем, навіть якщо їхні схеми відрізняються, що є особливо актуальним для організацій, котрі працюють з великими обсягами розподілених даних.

- RDF є основою для побудови семантичного вебу, де дані не лише зберігаються, а й мають чітко визначене значення.

- RDF дозволяє поступово розширювати модель даних, додаючи нові триплети без необхідності змінювати вже існуючу структуру, і таким чином забезпечувати гнучкість при розвитку інформаційних систем.

- Використання RDF забезпечує уніфікований підхід до опису та обміну метаданими, що сприяє інтероперабельності між різними застосунками та платформами.

RDF знайшов практичне застосування у багатьох сферах діяльності, зокрема:

- для опису наукових публікацій, даних досліджень, навчальних ресурсів, сприяючи їхній доступності та повторному використанню.

- для опису адміністративних реєстрів і відкритих даних, які сприяють прозорості та зручності пошуку інформації громадянами.

- для створення інтелектуальних систем пошуку, які враховують не лише ключові слова, а й змістовні зв'язки між об'єктами.

- для опису властивостей та зв'язків між пристроями, що забезпечує їхню взаємодію та інтеграцію у спільні рішення.

1.2.3 Extensible Markup Language (XML)

XML (Extensible Markup Language) – це гнучка мова розмітки, розроблена для зберігання, опису та передачі структурованих даних у текстовому форматі. Стандарт XML був затверджений World Wide Web Consortium (W3C) у 1998 році і швидко здобув популярність як універсальний формат обміну інформацією між різними системами завдяки своїй простоті, читабельності та можливості створювати власні теги, що відповідають конкретним потребам користувачів.

XML має такі основні характеристики:

- Розширюваність

XML не має фіксованого набору тегів – це дає змогу користувачам створювати власні теги для опису будь-яких типів даних, включно з метаданими, забезпечуючи гнучкість у моделюванні інформаційних структур.

- Структурованість

XML підтримує ієрархічну структуру даних, даючи можливість описувати складні об'єкти та їх взаємозв'язки. Кожен елемент може містити вкладені елементи, атрибути та текст, що надає деталізоване представлення інформації.

- Універсальність і підтримка

XML підтримується більшістю мов програмування, баз даних, систем обробки документів і веб-технологій.

Стандарти метаданих XML:

Завдяки своїй гнучкості та структурованості, XML став основою для створення численних стандартів метаданих, які широко використовуються у цифрових бібліотеках, архівах, видавничій справі та наукових дослідженнях.

Найбільш відомі:

- METS (Metadata Encoding and Transmission Standard)

Стандарт, що визначає XML-схему для упаковки та обміну метаданими цифрових об'єктів. Він включає описи структури об'єкта, адміністративних, технічних та описових метаданих і широко застосовується в архівних та бібліотечних системах для опису складних цифрових колекцій, забезпечуючи єдину точку доступу до різномірної інформації.

- MODS (Metadata Object Description Schema)

Легкий XML-стандарт, що був розроблений Бібліотекою Конгресу США. Він базується на елементах MARC (Machine-Readable Cataloging) і призначений для опису бібліографічних метаданих із збереженням гнучкості та простоти. MODS часто використовується для каталогізації та обміну бібліографічною інформацією.

- TEI (Text Encoding Initiative)

Стандарт для кодування текстів гуманітарних наук, який дозволяє детально описувати структуру, зміст і контекст текстових документів та широко застосовується у лінгвістиці, історії, літературознавстві та інших гуманітарних дисциплінах для створення цифрових текстових архівів.

Практичні застосування XML у сфері метаданих:

XML-файли легко передавати між різними інформаційними системами, забезпечуючи сумісність навіть при різних технологічних платформах. Це сприяє інтеграції даних у складних інформаційних екосистемах.

Багато програмних продуктів використовують XML для збереження налаштувань, полегшуючи їхню модифікацію, автоматизацію та підтримку.

XML дозволяє описувати не лише базові метадані, а й складні ієрархічні структури – це критично важливо для цифрових архівів, наукових публікацій, електронних книг та інших складних цифрових об'єктів.

Впровадивши XML-стандарти метаданих, з'являються деякі переваги, серед них:

- Структуровані метадані, що представлені у форматі XML, полегшують пошук, фільтрацію та навігацію по інформаційних ресурсах, таким чином підвищуючи ефективність роботи користувачів.

- Стандарти на базі XML забезпечують уніфікований формат обміну даними між різними системами, що сприяє їхній сумісності та спрощує інтеграцію.

- XML-документи легко зберігати, архівувати та відновлювати. Це є важливим для цифрових бібліотек, архівів та наукових репозитаріїв, адже для них збереження інформації на довгі терміни має критичне значення.

- XML-файли легко обробляються програмними засобами, що дозволяє створювати складні робочі процеси для управління метаданими, включаючи автоматичне генерування, валідацію та трансформацію даних.

1.3 Формати документів та інструменти для роботи з метаданими

1.3.1 Файл PDF і його метадані

PDF - це універсальний формат для представлення електронних документів, розроблений компанією Adobe у 1992 році. Його основна перевага полягає у збереженні структури, форматування та зовнішнього вигляду документа незалежно від операційної системи, програмного чи апаратного забезпечення. У 2008 році формат PDF було стандартизовано як відкритий стандарт ISO 32000-1, що сприяло його широкому впровадженню у науці, видавничій справі, юридичній сфері та архівуванні.

Метадані PDF - це додаткова інформація про документ, яка зберігається у його властивостях і служить для полегшення організації, пошуку, ідентифікації та автоматизованої обробки документів у цифрових архівах, бібліотеках, корпоративних системах управління документами. Метадані можуть використовуватися для індексації, каталогізації, контролю версій, забезпечення відповідності стандартам доступності, а також для управління правами доступу та авторськими правами.

Таблиця 1.1.

Основні поля метаданих у PDF

Назва поля	Опис поля
Title	Заголовок документа.
Author	Ім'я особи або організації, що створила документ.
Subject	Короткий опис змісту документа.
Keywords	Терміни, що описують основні теми документа.
Creation Date	Дата та час створення документа.
Modification Date	Дата та час останньої зміни документа.
Creator	Назва програмного забезпечення, за допомогою якого був створений документ.
Producer	Програма або бібліотека, що сформувала PDF.
Language	Мова вмісту документа.

Окрім основних полів, вказаних у табл. 1.1, PDF може містити й інші спеціалізовані поля, наприклад, інформацію про ліцензії, захист авторських прав, структуру документа для полегшення доступності тощо.

Інструментів для перегляду та редагування метаданих PDF існує багато, серед найвідоміших:

- Adobe Acrobat

Перегляд і редагування метаданих здійснюється через меню File – Properties - вкладка Description або за допомогою комбінації клавіш Ctrl + D.

- CommonLook PDF Validator

Спеціалізований інструмент для перевірки відповідності PDF-файлів стандартам доступності (PDF/UA), включно з коректністю метаданих.

- Foxit PDF Editor

Альтернативний редактор PDF, який також підтримує роботу з метаданими.

- Програмні бібліотеки

Для автоматизованої обробки метаданих часто використовують бібліотеки, наприклад, pikepdf і PyPDF2 для мови програмування Python або iText для мови Java, що дозволяють читати, змінювати та додавати метадані у PDF-файли.

Метадані PDF відіграють ключову роль у цифровому управлінні документами, забезпечуючи:

- Покращену пошукову здатність, адже завдяки індексації метаданих документи швидко знаходяться у великих архівах та репозитаріях.

- Автоматизацію робочих процесів - метадані дозволяють автоматизувати сортування, класифікацію та контроль версій документів.

- Підвищення доступності, бо структуровані метадані сприяють створенню доступних документів, що відповідають вимогам законодавства про інклюзію.

- Забезпечення юридичної безпеки, зберігаючи інформацію про авторські права, ліцензії та цифрові підписи.

1.3.2 Формат CSV як засіб зберігання метаданих

CSV - це один із найпоширеніших форматів для зберігання табличних даних завдяки своїй простоті, легкості обробки та широкій сумісності з різноманітними програмними засобами. У сфері управління метаданими наукових публікацій CSV-файли часто використовуються для зберігання структурованої інформації про документи, такої як назви публікацій, імена авторів, ключові слова, анотації, посилання, ідентифікатори, дані про організації та джерела, класифікаційні коди та інші атрибути.

Основні переваги формату CSV для метаданих:

- Простота структури

CSV є текстовим форматом, який легко створити, читати та редагувати навіть у звичайному текстовому редакторі. Формат складається з рядків, де кожен рядок містить поля, розділені комами або іншими роздільниками.

- Широка підтримка інструментів

Формат CSV підтримується практично всіма табличними редакторами, такими як Microsoft Excel, Google Sheets, LibreOffice Calc, а також численними бібліотеками для програмування (наприклад, pandas у Python, OpenCSV у Java).

- Легка інтеграція з базами даних

CSV-файли легко імпортуються та експортуються у більшість систем управління базами даних (СУБД), що робить їх зручним форматом для передачі даних між різними інформаційними системами.

- Компактність

Відсутність надлишкової структури дозволяє зберігати великі обсяги даних у відносно компактному вигляді, що оптимізує зберігання і передачу інформації.

Недоліки формату CSV для метаданих:

- Відсутність формального опису типів даних

Усі значення у CSV зберігаються як текст – це може ускладнити автоматичне визначення числових, дат чи логічних полів. Внаслідок цього можливі помилки при обробці даних, особливо у великих і складних наборах.

- Відсутність підтримки вкладеності та ієрархії

CSV є форматом для двовимірних таблиць і не підтримує вкладені структури, що обмежує його використання для зберігання складних структурованих метаданих, які часто мають ієрархічний характер.

- Проблеми з кодуванням і роздільниками

При неправильному кодуванні символів, неоднозначному використанні роздільників або відсутності єдиного стандарту для позначення пропущених значень можливі помилки при імпорті та експорті даних.

Для подолання вказаних проблем було розроблено стандарт CSV on the Web (CSVW), який дозволяє описувати структуру, типи даних і семантику CSV-файлів за допомогою окремого JSON-файлу метаданих. Цей підхід забезпечує чіткий опис схеми CSV, включно з типами полів, обмеженнями, зв'язками між даними, полегшену інтеграцію CSV-файлів у семантичний веб та інші інформаційні системи, а також автоматизацію обробки метаданих і підвищення якості даних. CSVW активно підтримується W3C і стає стандартом для використання CSV у складних інформаційних екосистемах, зокрема у науці, державних реєстрах та відкритих даних.

Стандартний CSV-файл, що використовувався при тестуванні створеної програми містить такі поля:

- Current number: номер випуску.
- URL: посилання на веб-сторінку публікації.
- Shortname: скорочена назва або посилання на публікацію.
- art_doi: цифровий ідентифікатор об'єкта (DOI) публікації.
- Full PDF: посилання на повний текст у форматі PDF.
- Reviews: рецензії або відгуки на публікацію.
- Authors: список авторів публікації.
- Institution: установи, з якими пов'язані автори.

- Abstract: анотація публікації.
- Keywords: ключові слова, що описують зміст публікації.
- References: список використаних джерел або посилань.

1.3.3 Бібліотеки Python для роботи з PDF, CSV та метаданими

Для автоматизації додавання метаданих до наукових статей у форматі PDF у середовищі Python використовується низка бібліотек, кожна з яких виконує окрему функціональність у процесі обробки даних.

Pikepdf – це потужна бібліотека для роботи з PDF-файлами, що дозволяє читати, змінювати та зберігати PDF-документи, а також керувати як класичними метаданими (назва, автор, тема, ключові слова), так і XMP-метаданими. Pikepdf підтримує додавання, редагування та видалення метаданих, що робить її важливим інструментом для організації та збагачення інформації документа. Крім того, бібліотека дозволяє об'єднувати, розділяти, шифрувати PDF-файли, а також відновлювати пошкоджені документи, що є корисним для комплексної обробки наукових публікацій.

Pandas – популярна бібліотека для аналізу та обробки табличних даних. Вона забезпечує зручну роботу з CSV-файлами та дозволяє ефективно обробляти дані у структурі DataFrame, що робить опрацювання інформації гнучким та масштабованим для великої кількості публікацій. Pandas широко використовується для підготовки, фільтрації й трансформації метаданих перед їх інтеграцією у PDF-документи.

Lxml – бібліотека для роботи з XML-документами, яка надає повноцінну підтримку простору імен та можливість створення валідних XMP-блоків метаданих у форматі RDF. Це дає можливість користувачу ефективно створювати, змінювати й перевіряти структуру XMP-метаданих для PDF-файлів.

Re – модуль для роботи з регулярними виразами, який необхідний для очищення текстових полів перед записом у PDF-метадані: видалення HTML-тегів, витягування числових індексів, створення структурованих заголовків статей тощо. Модуль re дозволяє здійснювати пошук, заміну та розділення

тексту за шаблонами, що є особливо корисним для попередньої обробки неструктурованої інформації.

Висновки до розділу 1

1) Метадані є критично важливою складовою цифрових документів і забезпечують структуровану інформацію про зміст, авторство, тематику, дату створення, мову, ключові слова тощо. Вони підвищують пошукову доступність документів, сприяють індексації у базах даних і дозволяють організувати документи за змістовими критеріями.

2) Інформація, яка потрапляє у метадані, часто потребує очищення від HTML-тегів, зайвих лапок або нумерації, нормалізації авторів, усунення дублікатів у ключових словах та інше.

3) Для роботи з метаданими у Python використовуються спеціалізовані бібліотеки. У межах реалізованої програми було використано `pikepdf` для відкриття та редагування PDF-файлів, `lxml` для створення XML-структури XMP-метаданих і `pandas` для обробки CSV-таблиць з даними про статті.

РОЗДІЛ 2

АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ

2.1 Огляд програмних засобів

2.1.1 PyPDF2

PyPDF2 - одна з найпопулярніших бібліотек для роботи з PDF-файлами в мові програмування Python. Вона надає широкий спектр функцій для маніпуляції PDF-документами, що робить її зручною для автоматизації обробки документів, витягування інформації та редагування.

Основні можливості PyPDF2:

- Зчитування та витягування тексту з PDF-файлів, що дозволяє отримувати вміст для подальшої обробки, аналізу або індексації.
- Об'єднання кількох PDF-документів в один, розділення великих PDF на менші частини, а також обертання сторінок для коректного відображення.
- Додавання, зміну або видалення метаданих документів, що корисно для організації, каталогізації та пошуку файлів.
- Шифрування PDF для захисту вмісту, а також дешифрування захищених паролем файлів.

Переваги PyPDF2:

- Простота використання для базових операцій з PDF.
- Наявність детальної документації та великої спільноти користувачів.
- Підтримка багатьох стандартних функцій роботи з PDF.

Недоліки PyPDF2:

- Вразливість при роботі зі складними PDF-структурами, такими як багаторівневі таблиці, нестандартне форматування або інтерактивні елементи.
- Можливі помилки або втрата даних при обробці таких файлів.

2.1.2 pdfwr

pdfwr - це Python-бібліотека для читання, запису та маніпуляції PDF-файлами, яка відзначається гнучкістю у роботі з внутрішньою структурою документів.

Основні функції pdfwr:

- Зчитування та запис PDF-документів з повним доступом до їхньої внутрішньої структури, що дозволяє детально модифікувати документи.
- Об'єднання, розділення, обертання сторінок, а також модифікація метаданих.
- Створення PDF-документів з нуля при інтеграції з бібліотекою ReportLab, що дає змогу комбінувати існуючі PDF з новоствореним вмістом.

Переваги pdfwr:

- Відкритий вихідний код і можливість вільного використання.
- Простота у використанні для базових і середньо складних завдань.
- Можливість працювати автономно або у зв'язці з ReportLab для розширеного створення PDF.

Недоліки pdfwr:

- Менш активна підтримка та оновлення порівняно з іншими бібліотеками.
- Можливі проблеми з сумісністю з новими версіями Python або специфічними форматами PDF.

2.1.3 PyMuPDF

PyMuPDF, також відома як fitz – це високопродуктивна бібліотека для обробки PDF та інших форматів документів (серед них XPS, OpenXPS, EPUB), яка забезпечує широкий набір функцій для роботи з документами.

Основні можливості PyMuPDF:

- Швидке і точне витягування тексту, зображень і графічних елементів з PDF-файлів.

- Доступ до метаданих, підтримка анотацій, закладок, форм і інших інтерактивних елементів.
- Підтримка різних форматів документів, що розширює сферу застосування бібліотеки.

Переваги PyMuPDF:

- Висока швидкість обробки великих документів.
- Широкий функціонал для аналізу, редагування та маніпуляції PDF.
- Підтримка інтерактивних елементів, що робить її корисною для складних робочих процесів.

Недоліки PyMuPDF:

- Обмежена підтримка XMP-метаданих без використання додаткових зовнішніх бібліотек.
- Для деяких типів метаданих може знадобитися інтеграція з іншими інструментами.

2.2 Порівняльний аналіз

Порівняльний аналіз оформлено у вигляді наступної таблиці:

Таблиця 2.1.

Порівняльна таблиця розглянутих інструментів

Критерій	PyPDF2	pdfcrowd	PyMuPDF
Функціональні можливості	Зчитування та витягування тексту; об'єднання, розділення, обертання сторінок; додавання і зміна метаданих; шифрування і дешифрування PDF	Зчитування та запис PDF; об'єднання, розділення, зміна PDF; інтеграція з ReportLab для створення PDF з нуля; обертання сторінок	Витяг тексту і зображень; рендеринг сторінок у зображення; підтримка анотацій і закладок; редагування метаданих PDF та XML; об'єднання, розділення, обертання сторінок; шифрування, вбудовування шрифтів і зображень

Продовження табл. 2.1.

Переваги	Простота використання для базових операцій і підтримка багатьох стандартних функцій	Гнучкість у роботі з PDF-структурами; можливість створення PDF з нуля через ReportLab; відкритий код, швидкий парсер	Висока швидкість обробки; широкий спектр функцій для аналізу та маніпуляції PDF; збереження структури документа при витяганні тексту; підтримка роботи з метаданими PDF і XML; підтримка багатьох форматів документів і зображень
Недоліки	Обмежена підтримка складних структур PDF; не підтримує рендеринг сторінок; можливі проблеми з нестандартним форматуванням	Менш активна підтримка та оновлення; менша популярність і менш розвинена документація	Обмежена підтримка XMP-метаданих без додаткових бібліотек; вища складність API порівняно з PyPDF2 для початківців
Особливості	Підходить для простих операцій з PDF; не підтримує рендеринг сторінок	Може працювати як самостійно, так і у зв'язці з ReportLab; підтримка Unicode, гіперпосилань, вбудовування шрифтів	Підтримує рендеринг сторінок; підтримка витягання таблиць і складного контенту; підтримує роботу з багатьма форматами (PDF, XPS, EPUB тощо); можливість роботи з інтерактивними елементами PDF
Сфера застосування	Просте редагування, обробка тексту, базові операції з PDF	Створення та модифікація PDF-документів, особливо в поєднанні з ReportLab	Складний аналіз, витяг контенту, рендеринг, робота з інтерактивними PDF-документами, автоматизація обробки

За даними з табл. 2.1 можна зробити такі висновки про кожний з програмних засобів:

PyPDF2 є оптимальним вибором для базових операцій з PDF, коли потрібен простий інструмент для зчитування, об'єднання чи розділення файлів без складних вимог до структури документа.

pdfw підходить для більш гнучкої роботи з PDF, особливо якщо потрібно створювати документи з нуля або інтегруватися з ReportLab, але в такому випадку варто враховувати менш активний розвиток.

PyMuPDF – це найпотужніший і найшвидший інструмент серед трьох, ідеальний для складного аналізу, витягання тексту, зображень, таблиць, рендерингу сторінок і роботи з метаданими, проте вимагає більш глибокого розуміння API.

Висновки до розділу 2

1) Розглянуті бібліотеки PyPDF2, pdfw та PyMuPDF представляють різні підходи і рівні складності у роботі з PDF-документами на мові Python, що дає змогу обрати інструмент залежно від конкретних завдань і вимог.

2) Вибір конкретної бібліотеки слід здійснювати з урахуванням поставлених завдань: для простих операцій підійде PyPDF2, для створення і гнучкої модифікації PDF краще обрати pdfw, а для комплексної обробки, аналізу та рендерингу ліпшим варіантом буде PyMuPDF.

3) Загалом, всі три бібліотеки доповнюють одна одну, забезпечуючи широкий спектр можливостей для роботи з PDF у Python.

РОЗДІЛ 3

РОЗРОБКА ПРОГРАМИ НА PYTHON

3.1 Постановка задачі

У зв'язку зі значним обсягом наукових статей, які щороку публікуються у журналі «Functional Materials», виникла потреба в автоматизації процесу додавання метаданих до PDF-файлів. Ручне введення авторів, ключових слів, назв та анотацій є трудомістким, схильним до помилок і не масштабується на велику кількість документів.

Основна задача полягає в тому, аби розробити програмне рішення на Python, яке на основі табличного файлу CSV аналізує вміст полів (автори, ключові слова, коротка назва, анотація), очищує та уніфікує дані, додає метадані до відповідного PDF-файлу та підтримує стандартні метадані (PDF docinfo) та розширені XMP-метадані.

3.2 Опис алгоритму

Загальна схема роботи програми:

Крок 1: Вибір CSV-файлу - користувач вказує файл з метаданими.

Крок 2: Читання таблиці з використанням pandas, роздільник |.

Крок 3: Обробка кожного рядка:

- Побудова назви PDF-файлу.
- Витягування, очищення і форматування авторів і ключових слів.

Крок 4: Додавання метаданих:

- PDF docinfo: Title, Author, Subject, Keywords.
- XMP-метадані: через lxml формуються RDF-елементи, згідно з

Dublin Core.

Крок 5: Збереження оновленого PDF за тим самим шляхом.

Крок 6: Підрахунок успішно оновлених та неоновлених файлів.

3.3 Використані бібліотеки

Під час реалізації програми були використані такі бібліотеки мови Python:

1. os для роботи з файловою системою (отримання шляхів, перевірка існування файлів, з'єднання каталогів і назв файлів).
2. re - модуль регулярних виразів для очищення даних (видалення HTML-тегів, індексів авторів, нормалізації ключових слів тощо).
3. pandas для зручного читання та обробки вмісту CSV-файлу, що містить метадані для PDF-документів.
4. tkinter - стандартна бібліотека Python для створення GUI-інтерфейсів, використовується для виклику діалогових вікон відкриття CSV-файлу та вибору папки з PDF-файлами.
5. pikerpdf - бібліотека для читання, зміни та збереження PDF-файлів із можливістю редагування стандартних та XMP-метаданих.
6. lxml.etree для створення та форматування структури XMP-метаданих у вигляді XML-дерева з відповідними RDF-тегами.

Разом ці бібліотеки забезпечують автоматизацію процесу додавання структурованих метаданих до PDF-файлів відповідно до формату, прийнятого в наукових публікаціях.

3.4 Тестування програми

Методика тестування наведена на рисунках 3.1 – 3.15:

Крок 1: Підготовка вхідних даних

- Створено CSV-файл із метаданими:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	
1	Current number	URL	Shortname	art_doi	Full PDF	Reviews	Authors	Institution	Abstract	Keywords	References								
2	32-1	cont	E. Dolzhe	P. Konevs	L. Lytvyn	A. Volosh	SSI вЪльr	National / 60 Nauky	61001 Kh:	Ukraine	/ the therm	the optic	Kyropouk	optical ab	dislocatio	chemical	X-ray diffi	fracture t	Yole Deve
3	32-1	cont	Br	l	S.S. Srr	V.L. Karbi	V.Kh. Kasl	S.A. Nedil	S.I. Shulyr	N.A.Kurge	O.A. Puzk	L.I. Karbiv	A.P. Sorol	blvd. Acar	36	03142 Kyi	Ukraine	l	Br
4	32-1	cont	V.M. Mat	A.V. Koro	2</sup>		Zaporizh	69063	Ukraine	 <sup>36	Acaden	Kyiv	3142	Ukraine	l	caused by	has been	the nanoc	cubic latt
5	32-1	cont	GaAs sub	M.A.Shon	J.A.Razzal	14 Kh.	Alir	Urgench	220100	Uzbekista	10 August	Leipzig 04	Germany	n=(4f-7)B	respective	while the	heterostr	solid solu	hypotheri
6	32-1	cont	magnesi	Ye. O. Kor	O.V. Livits	I.I. Grynyi	D.K. Procl	Volodym	01601 Kyi	Ukraine	<l	78 Winstc	2094	Kyiv	Ukraine	<l	Peremoh	37	3056
7	32-1	cont	Muhamm	Mazin A.	. College of	University	Mosul city	Iraq³	<sup>Di	College of	University	Mosul city	Iraq	Supercor	0.1	0.2	0.3	0.4) and /	0.2) on th
8	32-1	cont	D.B. Hlusi	V.M. Volc	Magdebu	14770 Brz	Germany	25 Yarosl	61002 Kh:	Ukraine	<c	24a Archi	49000 Dn	Ukraine	l	given the	and the n	and these	
9	32-1	cont	V.B.Mura	V.E.Shelu	T.V.Khom	I.S.Marts	M.A.Vasil	V.V.Kremi	National / 3	Omelya	03142 Kyi	Ukraine	<l	National / 13	Pokrov	04070 Kyi	Ukraine	l	in the tem
10	32-1	cont	Yu. A. Zhe	V.O. Prosl	A.V. Dzhe	S.I. Rudne	2 Kyryp	ch 61002 Kh:	Ukraine	/ the result	the overv	and the e	the value	than that	since the	programr	Co-V coat	overpoter	
11	32-1	cont	V. Bilozer	O. Subboi	O. Volkov	S. Knyaze	V. Ryabos	V. Barann	V. Bobrov	Pf. Kalinik	S. Fesyuki	2 Kyryp	ch Kharkiv	61002	Ukraine		their phas	and hardi	
12	32-1	cont	Ye. Ph.	Yei Ph.	Рнь.V. Kamyans	l reaching	t the comp	micro-im	nickel-cla	PKHN-15	intensity	coefficient	L.L. Vovch	I.V. Ovsie	L.Yu. Mat	J. Questio	107	3 (2016).	
13	32-1	cont	dynamics	and temp	salicylald	and their	¹	^{1	2}	^{K 3}	<sup> <	State Scie	National / 60	Nauky	Kharkiv 6:	Ukraine	<l	V. N. Kara	
14	32-1	cont	L. Bohdar	S. Kyrii	O. Tymos	I. Pylppen	A. Burmai	G. Vasyli	T. Dontso	37	Prospect	Kyiv 0305	Ukraine	l	especially	is a ration	mechanic	and trans	
15	32-1	cont	Chunxia	C Xuan	Hou Bei	Liu Yu.	Shanxi	Yuncheng	China Fei	confocal	Fourier tr	universal	swelling p	thermog	chemical	mechanic	hygroscop	thermal s	
16	32-1	cont	A.V. Koro	2</sup>	<sup>	V.P. Kurb	R.Yu. Kor	Zaporizh	69063	Ukraine	 <sup>36	Acaden	Kyiv	3142	Ukraine	f	a blue shi	shape	
17	32-1	cont	controllec	46 Nauky	Kyiv	3028	Ukraine	l	the excita	while in tl	the single	induced b	Dirac	and topologic	single anc	Majorana	H. Yang	E. Derunc	
18	32-1	cont	O. G. Kuly	I. V. Hovo	H. S. Vlasi	A. L. Tatai	O. S. Kolo	60 Nauky	61072 Kh:	Ukraine	l	there is a	rapid	cost-eftec	selective	; we exami	while oth	including	
19	32-1	cont	M.Rucki	< R.V.Vovk	V.P.Nerut	D.Pieniak	V.O.Chysl	13 Akadei	20-950 Lu	Poland	<b	54 Stasiec	26-600 Ra	Poland	<b	4 Svobod	61022 Kh:	Ukraine	
20	32-1	cont	Guangdoi	Dongguar	China Ab	improved	crushing	: tin (Sn)	ar	Sn and Pb	the cope	and its re	respective	and the ri	B. H.	Science of	408	-2	
21	32-1	cont	A.I. Zhuk	Kharkov	Ukraine	Academic	1	Kharkov	61108 Uki	the Peclet	the activa	sublimati	effective	: Peclet nur	beryllium	N.Y. (201	: Yelliev	Yu	
22	32-1	cont	D.B. Hlusi	V.M. Volc	V.M. Rag	Magdebu	14770 Brz	Germany	25 Yarosl	61002 Kh:	Ukraine	<l	24a Archi	49000 Dn	Ukraine		indicating	bainite ar	
23	32-1	cont	O.V. Yelis	Yu.A. Gur	P.N. Zhm	V.D. Alek	L.O.Miros	STC вЪльr	60 Nauky	61072 Kh:	Ukraine	l	plastic sci	and their	demonstr	polystyre	polyphen	plastic sci	
24																			

Рис. 3.1 - Вигляд CSV-файлу з метаданими.

Файл складається з 23 рядків даних: перший рядок – заголовки, а починаючи з другого, наводяться метадані до PDF-файлів. Відповідно файл містить метадані для 22 наукових статей.

- Створено папку з відповідними PDF-файлами:

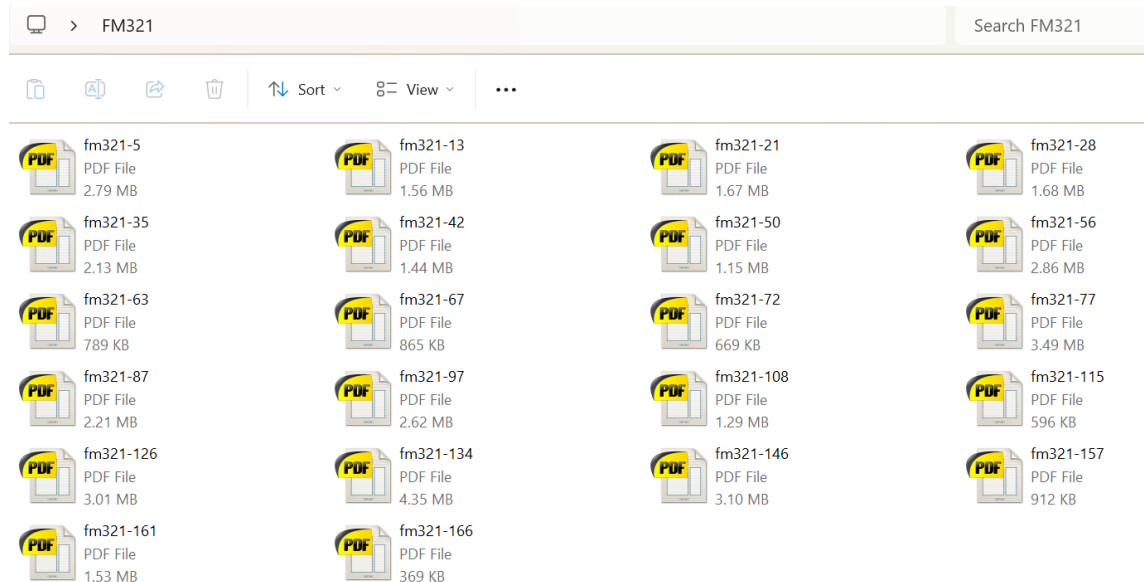


Рис. 3.2 - Папка з PDF-файлами.

Крок 2: Функціональне тестування

На цьому етапі:

- перевірялась правильність парсингу CSV-файлу.
- тестувались функції очистки HTML-тегів, ключових слів і авторів.
- оцінювалась коректність генерації XMP-метаданих.

Було обрано файл **fm321-63.pdf**. Відкривши його в Adobe Acrobat Reader, можна переглянути його метадані. Початково їх немає, це видно з наведеного нижче скріншота.

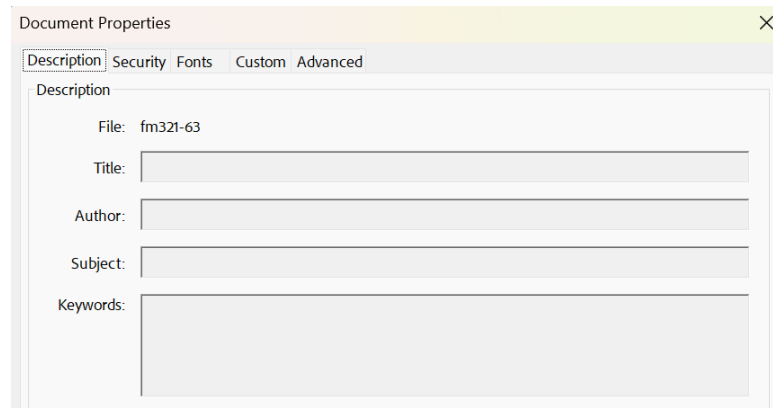


Рис. 3.3 - Метадані файлу fm321-63.pdf до запуску програми.

Запустивши програму в Visual Studio Code або будь-якому іншому середовищі, що підтримує Python, користувач побачить два вікна, у яких треба зробити вибір: спершу необхідно вибрати CSV-файл, а далі – папку з PDF-файлами.

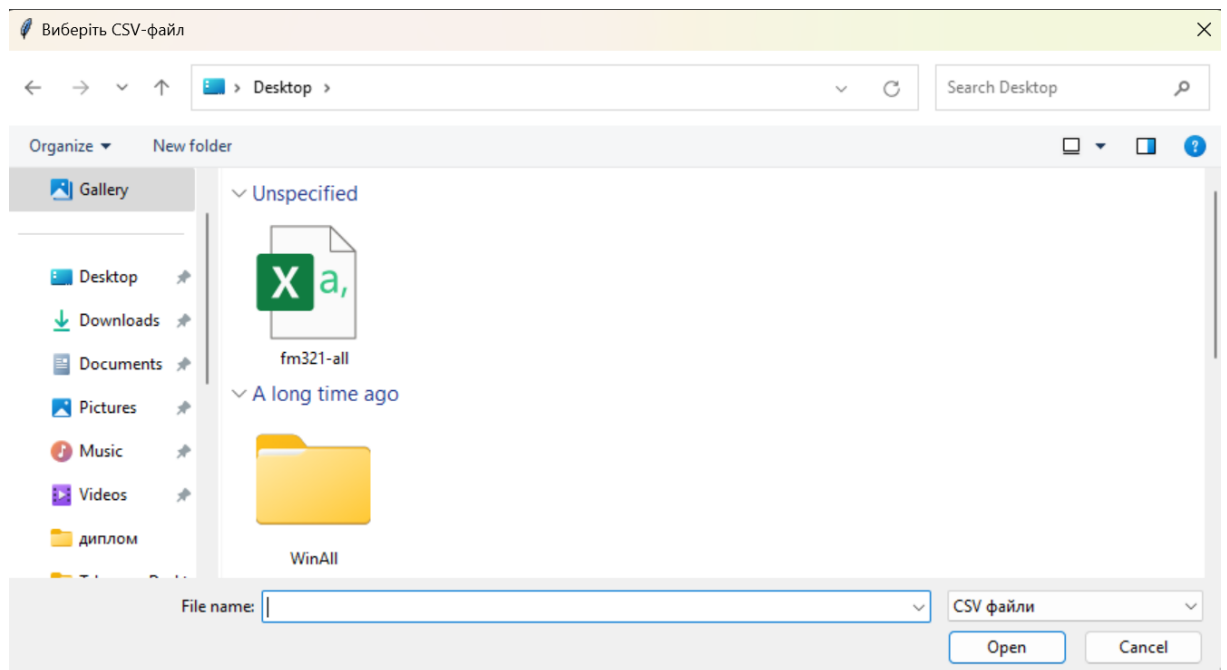


Рис. 3.4 - Вибір CSV-файлу.

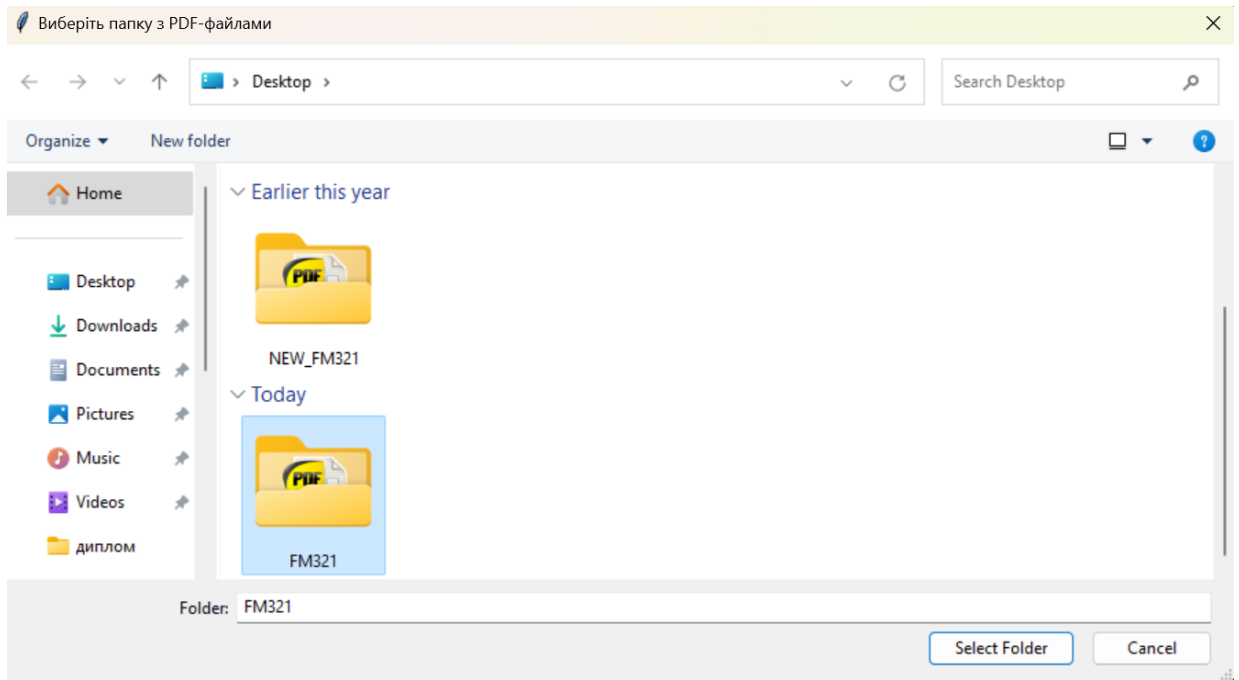


Рис. 3.5 - Вибір папки з PDF-файлами.

У нормальному випадку, тобто коли обрані файли мають правильні назви та формати, у консолі з'явиться повідомлення про успішність роботи програми.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\BVA> & C:/Users/BVA/python/python.exe c:/Users/BVA/Desktop/312-all_test.py
Успішно оновлено: fm321-5.pdf
Успішно оновлено: fm321-13.pdf
Успішно оновлено: fm321-21.pdf
Успішно оновлено: fm321-28.pdf
Успішно оновлено: fm321-35.pdf
Успішно оновлено: fm321-42.pdf
Успішно оновлено: fm321-50.pdf
Успішно оновлено: fm321-56.pdf
Успішно оновлено: fm321-63.pdf
Успішно оновлено: fm321-67.pdf
Успішно оновлено: fm321-72.pdf
Успішно оновлено: fm321-77.pdf
Успішно оновлено: fm321-87.pdf
Успішно оновлено: fm321-97.pdf
Успішно оновлено: fm321-108.pdf
Успішно оновлено: fm321-115.pdf
Успішно оновлено: fm321-126.pdf
Успішно оновлено: fm321-134.pdf
Успішно оновлено: fm321-146.pdf
Успішно оновлено: fm321-157.pdf
Успішно оновлено: fm321-161.pdf
Успішно оновлено: fm321-166.pdf

Результат:
Успішно оновлено: 22
Помилка: 0
```

Рис. 3.6 - Результат виконання програми.

Тепер можна знову подивитись метадані PDF-файлів та впевнитись, що метадані додалися, і що вони відповідають статті.

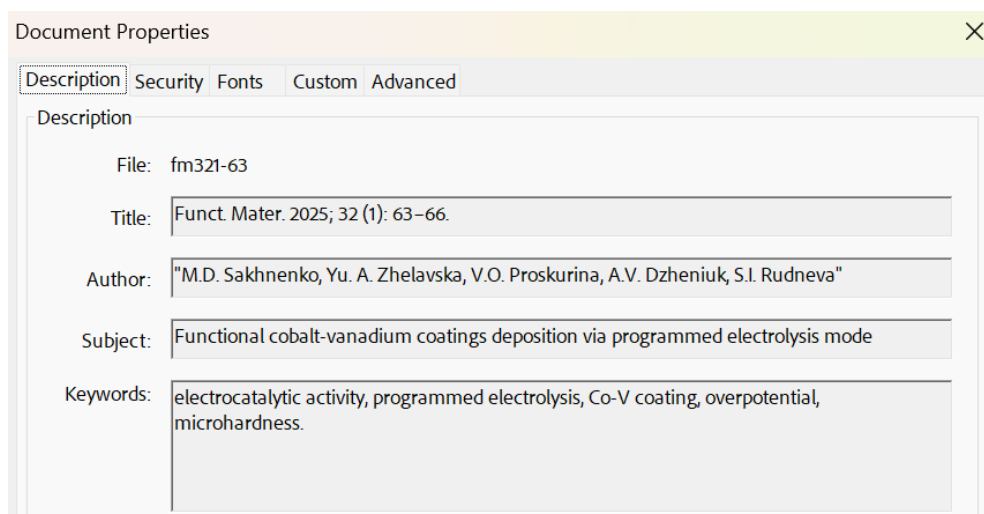


Рис. 3.7 - Метадані файлу fm321-63.pdf після запуску програми.

ISSN1027-5495. Functional Materials, 32, No.1 (2025), p. 63-66
doi:http://dx.doi.org/10.15407/fm32.01.63 © 2025 — STC "Institute for Single Crystals"

Functional cobalt-vanadium coatings deposition via programmed electrolysis mode

*M.D. Sakhnenko, Yu. A. Zhelavska, V.O. Proskurina,
A.V. Dzheniuk, S.I. Rudneva*

National Technical University "Kharkiv Polytechnic Institute",
2 Kyrpychova St., 61002 Kharkiv, Ukraine

Received October 15, 2014

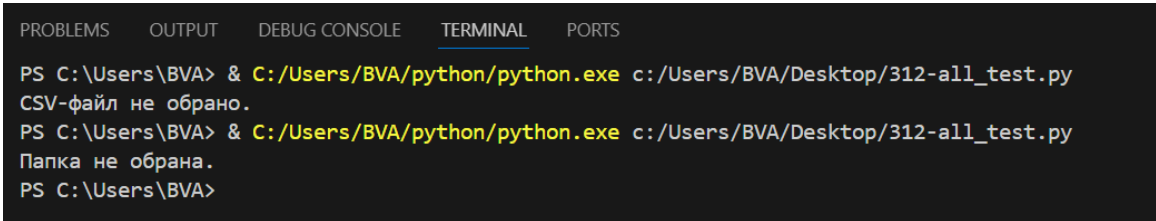
The electrocatalytic activity of a cobalt-vanadium alloy coating in the hydrogen evolution reaction was investigated. The cobalt-vanadium coating is electrodeposited from a complex citrate electrolyte via pulse electrolysis modes. According to the results of the X-ray dispersive spectrometry, the resulting coating contains 0.2–1.0 mass % vanadium. It is found that on a cathode with a Co-V coating, the overvoltage in the hydrogen evolution reaction is 0.15-0.2 V lower, and the exchange current is higher than on a cobalt coated electrode. As the vanadium content in the alloy increases, the value of the hydrogen evolution overvoltage decreases. This indicates the electrocatalytic activity of the vanadium-containing materials in the hydrogen evolution reaction. The microhardness of the Co-V coating is higher, than that of the cobalt coating. Electrodes coated with Co-V can be recommended as cathodes for the electrochemical production of hydrogen, since the energy consumption of this process is reduced by 10-15 %.

Keywords: electrocatalytic activity, programmed electrolysis, Co-V coating, overpotential, microhardness.

Рис. 3.8 – Фрагмент статті для перевірки відповідності доданих метаданих.

Крок 3: Тестування на виняткові ситуації

- Користувач закрив вікно з вибором CSV-файлу або папки з PDF-файлами:



```

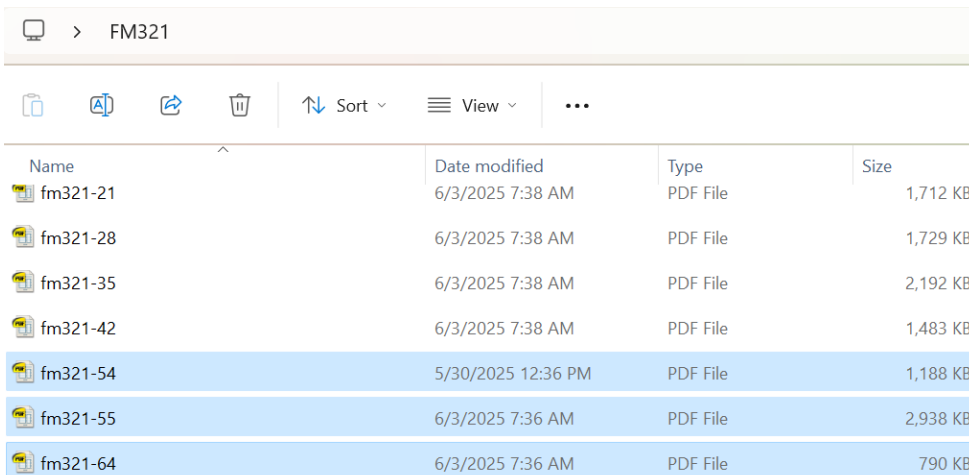
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\BVA> & C:/Users/BVA/python/python.exe c:/Users/BVA/Desktop/312-all_test.py
CSV-файл не обрано.
PS C:\Users\BVA> & C:/Users/BVA/python/python.exe c:/Users/BVA/Desktop/312-all_test.py
Папка не обрана.
PS C:\Users\BVA>

```

Рис. 3.9 - Користувач не обрав файли і програма завершилась.

- Відсутність PDF-файлу.

Для цього тесту було змінено назви декількох PDF-файлів у вихідній папці.



Name	Date modified	Type	Size
fm321-21	6/3/2025 7:38 AM	PDF File	1,712 KB
fm321-28	6/3/2025 7:38 AM	PDF File	1,729 KB
fm321-35	6/3/2025 7:38 AM	PDF File	2,192 KB
fm321-42	6/3/2025 7:38 AM	PDF File	1,483 KB
fm321-54	5/30/2025 12:36 PM	PDF File	1,188 KB
fm321-55	6/3/2025 7:36 AM	PDF File	2,938 KB
fm321-64	6/3/2025 7:36 AM	PDF File	790 KB

Рис. 3.10 - Три файли в папці змінено.

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\BVA> & C:/Users/BVA/python/python.exe c:/Users/BVA/Desktop/312-all_test.py
Успішно оновлено: fm321-5.pdf
Успішно оновлено: fm321-13.pdf
Успішно оновлено: fm321-21.pdf
Успішно оновлено: fm321-28.pdf
Успішно оновлено: fm321-35.pdf
Успішно оновлено: fm321-42.pdf
Файл не знайдено: fm321-50.pdf
Файл не знайдено: fm321-56.pdf
Файл не знайдено: fm321-63.pdf
Успішно оновлено: fm321-67.pdf
Успішно оновлено: fm321-72.pdf
Успішно оновлено: fm321-77.pdf
Успішно оновлено: fm321-87.pdf
Успішно оновлено: fm321-97.pdf
Успішно оновлено: fm321-108.pdf
Успішно оновлено: fm321-115.pdf
Успішно оновлено: fm321-126.pdf
Успішно оновлено: fm321-134.pdf
Успішно оновлено: fm321-146.pdf
Успішно оновлено: fm321-157.pdf
Успішно оновлено: fm321-161.pdf
Успішно оновлено: fm321-166.pdf

Результат:
Успішно оновлено: 19
Помилка: 3

```

Рис. 3.11 - Потрібні PDF-файли відсутні в обраній папці.

Програма повідомляє, що під час роботи сталося 3 помилки і деякі файли не були оброблені.

- Відсутність деяких полів у CSV.

Для цього тесту було видалено поле «Current number» для перших двох статей.

	A	B	C	D	E	F	G	H	I	J		
1	Current number	URL	Shortname	art_ doi	Full PDF	Reviews	Authors	Institution	Abstact	Keywords	References	
2	contents/32-1/5	Fu	E. Dolzhe	P. Konevs	L. Lytvyns	A. Volosh	SSI вЪнлr	National /	60 Nauky	61001 Kh	Ukraine /	
3	contents/32-1/13	Fu	Br)	S.S. Sn	V.L. Karbi	V.Kh. Kasi	S.A. Nedil	S.I. Shulyr	N.A.Kurga	O.A. Puzk	L.I. Karbiv
4	32-1	contents/32-1/	V.M. Mat	A.V. Koro	2</sup>		Zaporizhzh	69063	Ukraine	 <sup>	36 Acaden	Kyiv
5	32-1	contents/32-1/	GaAs sub	M.A.Shon	J.A.Razzal	14 Kh.Ali	Urgench	220100	Uzbekista	10 August	Leipzig 04	
6	32-1	contents/32-1/	magnesium	Ye. O. Kor	O.V. Livits	I.I. Gryny	D.K. Procl	Volodymy	01601 Kyi	Ukraine<	78 Winstc	
7	32-1	contents/32-1/	Muhamm	Mazin A.	College of	University	Mosul city	Iraq^{	3}D	College of	University	
8	32-1	contents/32-1/	D.B. Hlusi	V.M. Volc	Magdebu	14770 Bra	Germany	25 Yarosl	61002 Kh	Ukraine<	24a Archi	
9	32-1	contents/32-1/	V.B.Mura	V.E.Shelu	T.V.Khom	I.S.Martse	M.A.Vasil	V.V.Krem	National /	3 Omelya	03142 Kyi	

Рис. 3.12 - Видалення декількох полів з CSV-файлу.

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\BVA> & C:/Users/BVA/python/python.exe c:/Users/BVA/Desktop/312-all_test.py
Помилка читання CSV: No columns to parse from file

```

Рис. 3.13 - Повідомлення про помилку і завершення роботи програми.

- Некоректний роздільник

Для цього тесту у CSV-файлі роздільник | замінено на ; .

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Current number;URL;Shortname;art_doi;Full PDF;Reviews;Authors;Institution;Abstact;Keywords;References												
2	32-1;cont	E. Dolzhe	P. Konevs	L. Lytvyn	A. Volosh	SSI вЪьлг	National /	60 Nauky	61001 Kh:	Ukraine;	A the therm	the optica	Kyropoulk opt
3	32-1;cont	Br	l);S.S. Sm	V.L. Karbi	V.Kh. Kasi	S.A. Nedil	S.I. Shulyr	N.A.Kurge	O.A. Puzk	L.I. Karbiv	A.P. Sorol	blvd. Аса	36 031
4	32-1;cont	V.M. Mat	A.V. Koro	2</sup>	> Zaporizh	69063	Ukraine	 <sup	36 Acaden	Kyiv	3142	Ukraine;	T caused by has
5	32-1;cont	GaAs sub:	M.A.Shon	J.A.Razzal	14 Kh.Ali	Urgench	220100	Uzbekista	10 August	Leipzig 04	Germany;	n=(4Г·7)B	respectiv wh
6	32-1;cont	magnesi	Ye. O. Kor	O.V. Livits	I.I. Grynyi	D.K. Procl	Volodym	01601 Kyi	Ukraine<	78 Winstc	2094	Kyiv	Ukraine<
7	32-1;cont	Muhamm	Mazin A.	College of	University	Mosul city	Iraq^{	3}D	College of	University	Mosul city	Iraq	;Supercon
8	32-1;cont	D.B. Hlusl	V.M. Volc	Magdebu	14770 Bra	Germany	25 Yarosl	61002 Kh:	Ukraine<	24a Archi	49000 Dn	Ukraine;	T given the anc
9	32-1;cont	V.B.Mura	V.E.Shelu	T.V.Khom	I.S.Marts	M.A.Vasil	V.V.Kremi	National /	3 Omelya	03142 Kyi	Ukraine<	National /	13 Pokrov: 04C

Рис. 3.14 - Зміна роздільника.

```
143 df = pd.read_csv(csv_path, delimiter="|") # вказати роздільник
144 except Exception as e:
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\BVA> & C:/Users/BVA/python/python.exe c:/Users/BVA/Desktop/312-all_test.py
Помилка читання CSV: No columns to parse from file
```

Рис. 3.15 - Повідомлення про помилку і завершення роботи програми.

Висновки до розділу 3

Було реалізовано програму для автоматизованого додавання метаданих до наукових PDF-статей, використовуючи інформацію з CSV-файлу. Під час розробки були розглянуті вимоги до метаданих відповідно до стандартів XMP та Dublin Core, а також проведено структурний аналіз необхідних полів (назва, автори, анотація, ключові слова тощо).

Програма реалізована мовою Python з використанням таких бібліотек, як pikepdf, lxml, pandas, tkinter та інших. Особливу увагу приділено:

- автоматичному очищенню і нормалізації даних (видаленню HTML-тегів та індексів авторів);
- врахуванню можливих помилок, таких як відсутність файлів чи невідповідність назв.

У результаті тестування було підтверджено працездатність програми в типових і виняткових ситуаціях. Програма успішно виконує поставлену задачу та може бути адаптована до інших форматів зберігання метаданих у видавничій галузі.

РОЗДІЛ 4

ПІДБИТТЯ ПІДСУМКІВ І ПЕРСПЕКТИВИ РОЗВИТКУ

4.1 Робота програми

Розроблена програма дозволяє автоматично додавати метадані до PDF-документів наукових статей на основі даних із CSV-файлу. Програма успішно виконує такі основні функції:

- Зчитування та обробка структурованих даних (назви статей, автори, ключові слова, анотації тощо);
- Обробка декількох документів у пакеті;
- Обробка нестандартних ситуацій (відсутність файлів, помилки структури тощо).

У результаті тестування підтверджено, що програма стабільно працює з більшістю наукових публікацій у PDF-форматі, забезпечує коректне збереження введених метаданих та відповідає вимогам до цифрових бібліотек.

4.2 Можливості вдосконалення

Незважаючи на те, що базова функціональність програми реалізована успішно, існують напрямки для подальшого її вдосконалення:

A. Підтримка інших форматів зберігання метаданих

Можна розширити програму для обробки JSON, XML або Excel-файлів як джерел метаданих та імпорту метаданих з інтерфейсів типу CrossRef, ORCID тощо.

B. Інтеграція з базами даних

Підключення до баз даних дозволить:

- зберігати інформацію про оброблені документи;
- вести історію змін або реєстр метаданих;
- синхронізувати з редакційною системою електронного журналу чи сайтом.

C. Автоматичне вилучення метаданих з PDF

Додатково можлива реалізація зворотного процесу, тобто вилучення наявних метаданих з PDF-документів для перевірки, аналізу або міграції до інших форматів.

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було реалізовано комплексне рішення для автоматизації обробки метаданих наукових публікацій у форматі PDF. Досягнута мета полягала у створенні комп'ютерної моделі, яка дозволяє швидко, ефективно та безпомилково додавати структуровану інформацію до великої кількості документів із зовнішнього джерела, а саме CSV-файлу.

У результаті виконання роботи отримано наступні основні результати:

1) Проаналізовано поняття та типи метаданих, їх роль у системах електронного документообігу, видавничих платформах і цифрових репозитаріях. Розглянуто основні міжнародні стандарти представлення метаданих - Dublin Core, RDF та XMP, а також підходи до їх реалізації у PDF-документах.

2) Вивчено формати даних PDF та CSV, визначено їхні особливості у контексті збереження та обміну інформацією. Було проаналізовано, які саме метадані можуть зберігатися в PDF-файлах, і в яких структурах (стандартні поля docinfo і XMP).

3) Розроблено програмне забезпечення мовою Python, яке реалізує автоматизоване зчитування метаданих з CSV-файлу, їх попередню обробку (очищення HTML-тегів, нормалізація авторів і ключових слів), а також додавання до PDF-документів як у вигляді стандартних, так і XMP-метаданих.

4) У розробці програми використано набір спеціалізованих бібліотек Python, таких як pikepdf для обробки PDF, pandas для зчитування CSV і роботи з таблицями, re - регулярні вирази для очищення тексту, lxml.etree для генерації XMP у форматі RDF/XML та tkinter для організації інтерактивного вибору файлів користувачем.

5) Реалізовано пакетну обробку файлів, що дозволяє автоматично обробляти десятки або сотні документів за один запуск програми. Таким

способом можна значно зменшити витрати часу й уникнути помилок при ручному редагуванні PDF-файлів.

6) Проведено тестування програми, яке показало стабільну роботу, стійкість до типових помилок, а також відповідність доданих метаданих до статті.

7) Сформульовано потенційні напрями розвитку проекту, зокрема розширення джерел вхідних даних (JSON, XML), інтеграція з базами даних і взаємодія з системами типу Open Journal Systems або CrossRef.

8) Розроблена комп'ютерна модель є масштабованою, адаптивною до різних форматів вхідних даних, і потенційно придатною для використання у наукових видавництвах, цифрових бібліотеках, інституційних репозитаріях та редакційних системах журналів з відкритим доступом.

Так, виконана кваліфікаційна робота демонструє володіння студенткою сучасними технологіями обробки даних, навичками програмування, знанням стандартів опису наукових ресурсів і здатністю до самостійного вирішення комплексного прикладного завдання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1) National Information Standards Organization (NISO). Understanding Metadata. 2017. URL: <https://www.niso.org/publications/understanding-metadata-2017> (Дата звернення: 28.05.2025).
- 2) Wikipedia. Metadata. 2025. URL: <https://en.wikipedia.org/wiki/Metadata> (Дата звернення: 28.05.2025).
- 3) Dublin Core Metadata Initiative. Dublin Core Metadata Element Set, Version 1.1. 2012. URL: <https://www.dublincore.org/specifications/dublin-core/dces/> (Дата звернення: 28.05.2025).
- 4) World Wide Web Consortium (W3C). Resource Description Framework (RDF): Concepts and Abstract Syntax. 2014. URL: <https://www.w3.org/TR/rdf-concepts/> (Дата звернення: 28.05.2025).
- 5) TechTarget. Resource Description Framework (RDF). 2024. URL: <https://www.techtarget.com/searcharchitecture/definition/Resource-Description-Framework-RDF> (Дата звернення: 29.05.2025).
- 6) Bray, T., Paoli, J., Sperberg-McQueen, C. M., Maler, E., Yergeau, F. Extensible Markup Language (XML) 1.0 (Fifth Edition). W3C Recommendation, 2008. URL: <https://www.w3.org/TR/xml/> (Дата звернення: 29.05.2025).
- 7) Wikipedia. PDF. 2024. URL: <https://en.wikipedia.org/wiki/PDF> (Дата звернення: 29.05.2025).
- 8) Library of Congress. PDF, Version 1.7 (ISO 32000-1:2008). 2023. URL: <https://www.loc.gov/preservation/digital/formats/fdd/fdd000277.shtml> (Дата звернення: 29.05.2025)
- 9) World Wide Web Consortium (W3C). CSV on the Web (CSVW) Primer. 2018. URL: <https://www.w3.org/TR/tabular-data-primer/> (Дата звернення: 30.05.2025)
- 10) World Wide Web Consortium (W3C). CSV on the Web (CSVW) Metadata Vocabulary and Metadata File. 2018. URL: <https://www.w3.org/TR/tabular-metadata/> (Дата звернення: 30.05.2025)

- 11) Python Software Foundation. pandas Documentation: IO Tools (CSV, Excel, HDF5, ...). 2024. URL: https://pandas.pydata.org/docs/user_guide/io.html#csv-text-files (Дата звернення: 30.05.2025)
- 12) Python Software Foundation. pikepdf Documentation. 2024. URL: <https://pikepdf.readthedocs.io/en/latest/> (Дата звернення: 31.05.2025)
- 13) Python Software Foundation. lxml Documentation. 2024. URL: <https://lxml.de/> (Дата звернення: 31.05.2025)
- 14) Python Software Foundation. re - Regular expression operations. 2024. URL: <https://docs.python.org/3/library/re.html> (Дата звернення: 31.05.2025)
- 15) Python Software Foundation. os - Miscellaneous operating system interfaces. 2024. URL: <https://docs.python.org/3/library/os.html> (Дата звернення: 31.05.2025).
- 16) Python Software Foundation. tkinter - Python interface to Tcl/Tk. 2024. URL: <https://docs.python.org/3/library/tkinter.html> (Дата звернення: 01.06.2025)
- 17) Real Python. Working with PDFs in Python: PyPDF2, pdfrw, and PyMuPDF. 2023. URL: <https://realpython.com/pdf-python/> (Дата звернення: 01.06.2025)
- 18) PyPDF2. PyPDF2 documentation. 2024. URL: <https://pypdf2.readthedocs.io> (Дата звернення: 01.06.2025).
- 19) pdfrw GitHub Repository. 2023. URL: <https://github.com/pmaupin/pdfrw> (Дата звернення: 02.06.2025)
- 20) PyMuPDF Documentation. 2024. URL: <https://pymupdf.readthedocs.io/en/latest/> (Дата звернення: 02.06.2025)

ДОДАТКИ

Додаток А

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Харківський національний університет імені В. Н. Каразіна

ННІ комп'ютерних наук та штучного інтелекту
Кафедра комп'ютерних систем і робототехніки
Рівень вищої освіти (освітньо-кваліфікаційний рівень) Бакалавр
Напрямок підготовки 15 – Автоматизація та приладобудування
Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри комп'ютерних систем та робототехніки

к. ф.-м. н., доц. ХРУСЛОВ М. М.
«02» жовтня 2024 року

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Боровкова Віра Андріївна

(прізвище, ім'я, по батькові студента)

1. Тема роботи Комп'ютерна модель системи аналізу та управління метаданими наукових публікацій

керівник роботи ХРУСЛОВ Максим Михайлович, к. ф.-м. н., доц., в.о. завідувача кафедри комп'ютерних систем та робототехніки

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затвержені наказом по університету № 4101-5/962 від 16 квітня 2025 року

2. Строк подання студентом роботи **30 травня 2025 року**

3. Перелік питань, які потрібно розробити:

- 1) Дослідити існуючі підходи до управління метаданими наукових публікацій у PDF-форматі та розробити оптимальну структуру метаданих. Спроекувати архітектуру комп'ютерної моделі для автоматизованої обробки цих метаданих.
- 2) Створити програмний засіб на Python для імпорту, інтеграції та нормалізації метаданих з CSV-файлів у PDF-документи.
- 3) Забезпечити надійність роботи програми з обробкою помилок та провести тестування розробленої системи на реальних даних.
- 4) Оформити повний пакет технічної документації, включаючи опис програмного виробу та методику випробувань. Підготувати та захистити кваліфікаційну роботу.

4. План роботи

№ з/п	Назви етапів роботи	Терміни виконання
1	Затвердження теми роботи з керівником	01.10.2024 – 07.10.2024
2	Аналіз предметної області та огляд існуючих рішень	08.10.2024 – 31.10.2024
3	Аналіз та пошук методичної літератури щодо систем управління метаданими	01.11.2024 – 15.11.2024
4	Розробка вимог до програмного виробу	16.11.2024 – 30.11.2024
5	Проектування структури метаданих та архітектури системи	01.12.2024 – 15.12.2024
6	Розробка алгоритмів імпорту, обробки та запису метаданих	16.12.2024 – 31.12.2024
7	Реалізація програмного коду (імпорт, обробка, запис метаданих)	01.01.2025 – 31.01.2025
8	Тестування та налагодження програмного виробу	01.02.2025 – 15.02.2025
9	Аналіз ефективності та перспектив удосконалення моделі	16.02.2025 – 28.02.2025
10	Підготовка програмної документації	01.03.2025 – 10.03.2025
11	Підготовка та оформлення звітних матеріалів	11.03.2025 – 31.03.2025
12	Оформлення пояснювальної записки та додатків	01.04.2025 – 15.04.2025
13	Представлення кваліфікаційної роботи керівнику та рецензенту	16.04.2025 – 30.05.2025

5. Дата видачі завдання *02 жовтня 2024 року.*

Студент


 підпис
В.А. Боровкова
ініціали, прізвище

Керівник роботи


 підпис
М. М ХРУСЛОВ.
ініціали, прізвище

Додаток Б

ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ ПРОГРАМНОГО ВИРОБУ
«КОМП'ЮТЕРНА МОДЕЛЬ СИСТЕМИ АНАЛІЗУ ТА УПРАВЛІННЯ
МЕТАДАНИМИ НАУКОВИХ ПУБЛІКАЦІЙ»

Назва розділу	Назва і зміст підрозділу
1. Введення	<p>1.1. Назва програмного виробу: Комп'ютерна модель системи аналізу та управління метаданими наукових публікацій.</p> <p>1.2. Галузь застосування: Інформаційні технології, автоматизація наукової діяльності, цифрові бібліотеки, дослідницькі установи, університети.</p>
2. Підстава для розробки	<p>2.1. Навчальний план за спеціальністю: Виконання кваліфікаційної роботи за спеціальністю 151 «Автоматизація та комп'ютерно-інтегровані технології».</p> <p>2.2. Завдання на кваліфікаційну роботу: Розробка програмного продукту для автоматизованого додавання, аналізу та управління метаданими наукових публікацій у форматі PDF.</p>
3. Призначення розробки	<p>3.1. Мета розробки: Створення програмного продукту, що автоматизує процес додавання метаданих до PDF-файлів наукових публікацій.</p> <p>3.2. Призначення програмного виробу: Оптимізація роботи з електронними архівами наукових публікацій, підвищення якості та швидкості обробки метаданих.</p> <p>3.3. Вихідні дані для розробки:</p> <ul style="list-style-type: none"> - Вимоги до функціоналу (імпорт CSV, обробка PDF, форматування метаданих) - Структура метаданих: поля Authors, Shortname, Keywords, Reviews, URL - Приклади вхідних файлів .csv і .pdf.
4. Технічні вимоги до програмного виробу	<p>4.1. Вимоги до функціональних характеристик:</p> <ul style="list-style-type: none"> - Графічний інтерфейс для вибору CSV-файлу та папки з PDF-файлами. - Читання CSV з роздільником.

5. Вимоги до програмної документації	Програмною документацією вважаються: 1) Справжнє технічне завдання (Додаток Б) 2) Програма та методика випробувань (Додаток В) 3) Опис програмного виробу (розділ 3 пояснювальної записки) 4) Текст програми (Додаток Г)
6. Техніко-економічні показники	1) Орієнтовна ефективність: скорочення часу на підготовку метаданих, підвищення точності. 2) Терміни розробки: 3 місяці. 3) Витрати: мінімальні, пов'язані лише з роботою розробника. 4) Порівняння: безкоштовність та відкритий код на відміну від комерційних аналогів.
7. Стадії і етапи розробки	1) Аналіз предметної області та огляд існуючих рішень – 1 тиждень 2) Розробка вимог до програмного виробу – 1 тиждень 3) Проектування структури метаданих та архітектури системи – 2 тижні 4) Реалізація програмного коду (імпорт, обробка, запис) – 3 тижні 5) Тестування та налагодження програмного виробу – 3 тижні 6) Підготовка програмної документації – 2 тижні 7) Підготовка пояснювальної записки та захист роботи
8. Порядок контролю і приймання	1) Перевірка ходу розробки керівником 1 раз на 3 тижні. 2) Випробування згідно з методикою на базі комп'ютерного класу. 3) Захист на засіданні атестаційної комісії 4) Пояснювальна записка – на паперових та електронних носіях.

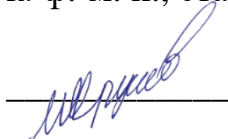
Виконавець

студентка групи КУ41 Боровкова Віра



Замовник

к. ф.-м. н., старший дослідник, доцент Хруслов Максим Михайлович



Додаток В**ПРОГРАМА І МЕТОДИКА ВИПРОБУВАНЬ ПРОГРАМНОГО ВИРОБУ
«КОМП'ЮТЕРНА МОДЕЛЬ СИСТЕМИ АНАЛІЗУ ТА УПРАВЛІННЯ
МЕТАДАНИМИ НАУКОВИХ ПУБЛІКАЦІЙ»****1. Об'єкт випробувань****1.1. Найменування випробуваного програмного виробу**

Комп'ютерна модель системи аналізу та управління метаданими наукових публікацій.

1.2 Область його застосування

Інформаційні технології, автоматизація наукової діяльності, цифрові бібліотеки, дослідницькі установи, університети.

2. Мета випробувань

Підтвердження відповідності функціональних, технічних та експлуатаційних характеристик розробленого програмного виробу вимогам, викладеним у технічному завданні, а також перевірка коректності додавання метаданих у PDF-файли на основі CSV-даних.

3. Загальні положення**3.1 Підстави для проведення випробувань**

Наказ про призначення атестаційної комісії для захисту кваліфікаційної роботи.

3.2 Місце і тривалість випробувань

Випробування проводяться на базі комп'ютерного класу кафедри протягом періоду підготовки та захисту кваліфікаційної роботи.

3.3 Обсяг випробувань

Приймальні випробування охоплюють перевірку функціональності, надійності, сумісності та відповідності технічним вимогам, визначеним у технічному завданні.

3.4 Організації, які беруть участь у випробуваннях

Студент, керівник роботи, Інститут сцинтиляційних матеріалів НАН України, атестаційна комісія.

4. Вимоги до програми або програмного виробу

Перевіряються наступні вимоги відповідно до технічного завдання:

- Графічний інтерфейс для вибору CSV-файлу та папки з PDF-файлами.
- Коректне зчитування CSV-файлів із роздільником «|».
- Правильне форматування та очищення полів метаданих (автори, ключові слова, заголовки).
- Додавання стандартних метаданих (Title, Author, Subject, Keywords) у PDF-файли.
- Генерація XMP-метаданих у форматі XML і їх коректне збереження у PDF.
- Стійкість до помилок: обробка відсутніх файлів, некоректних даних, винятків.
- Пакетна обробка файлів із виведенням звіту про успішні та помилкові операції.

5. Вимоги до програмної документації

Склад програмної документації, що подається на випробування, включає:

- Технічне завдання на розробку програмного виробу (Додаток Б до пояснювальної записки).
- Програму і методику випробувань (Додаток В до пояснювальної записки).
- Опис програмного виробу (розділ 3 пояснювальної записки).
- Текст програми (Додаток Г до пояснювальної записки).

6. Засоби і порядок випробувань

6.1 Засоби випробувань

- Технічні засоби: персональний комп'ютер з ОС Windows 11, встановлений Python 3.12.
- Програмні засоби: бібліотеки Python (pandas, pikepdf, lxml, tkinter).
- Вхідні дані: CSV-файли з метаданими, набір PDF-файлів для обробки.
- Вихідний код.

6.2 Порядок проведення випробувань

Випробування проводяться у два етапи:

1-й етап – ознайомчий:

- Перевірка комплектності програмної документації згідно з технічним завданням.

- Перевірка наявності необхідних технічних і програмних засобів.

2-й етап – власне випробування:

- Перевірка відповідності функціональних характеристик програми вимогам ТЗ.

- Тестування коректності імпорту CSV, обробки та додавання метаданих у PDF.

- Перевірка стійкості до помилок (відсутність файлів, некоректні дані).

- Оцінка зручності користувацького інтерфейсу.

- Верифікація звітності про виконані операції.

Методика проведення:

- Запуск програми, вибір вхідних файлів і папок через GUI.

- Виконання пакетної обробки тестових даних.

- Аналіз результатів: перевірка метаданих у PDF за допомогою сторонніх засобів (наприклад, Adobe Acrobat Reader).

- Фіксація результатів тестування, виявлення та усунення помилок.

7. Критерії приймання програмного виробу

- Програма виконує всі функції, визначені в технічному завданні.
- Метадані коректно додаються до PDF-файлів у стандартному та ХМР-форматах.
- Відсутність критичних помилок під час роботи з різними вхідними даними.
- Програма має зручний інтерфейс для користувача.
- Документація повна і відповідає вимогам.

Програмний виріб вважається таким, що пройшов випробування, якщо всі вищезазначені критерії виконані.

Для проведення випробувань пропонується тест 1, тест 2 і тест 3.

Тест 1: У обраному CSV-файлі є метадані для потрібного PDF-файлу, тож вони успішно додаються і відображаються.

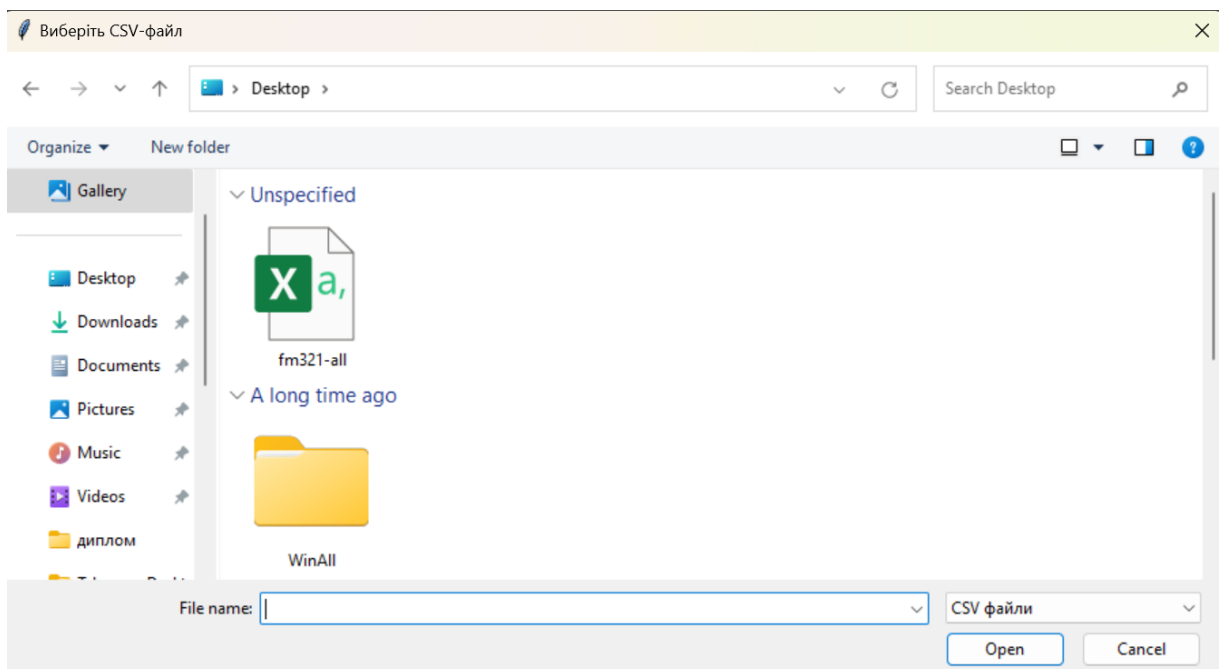


Рис. В1 - Тест 1, вибір CSV-файлу.

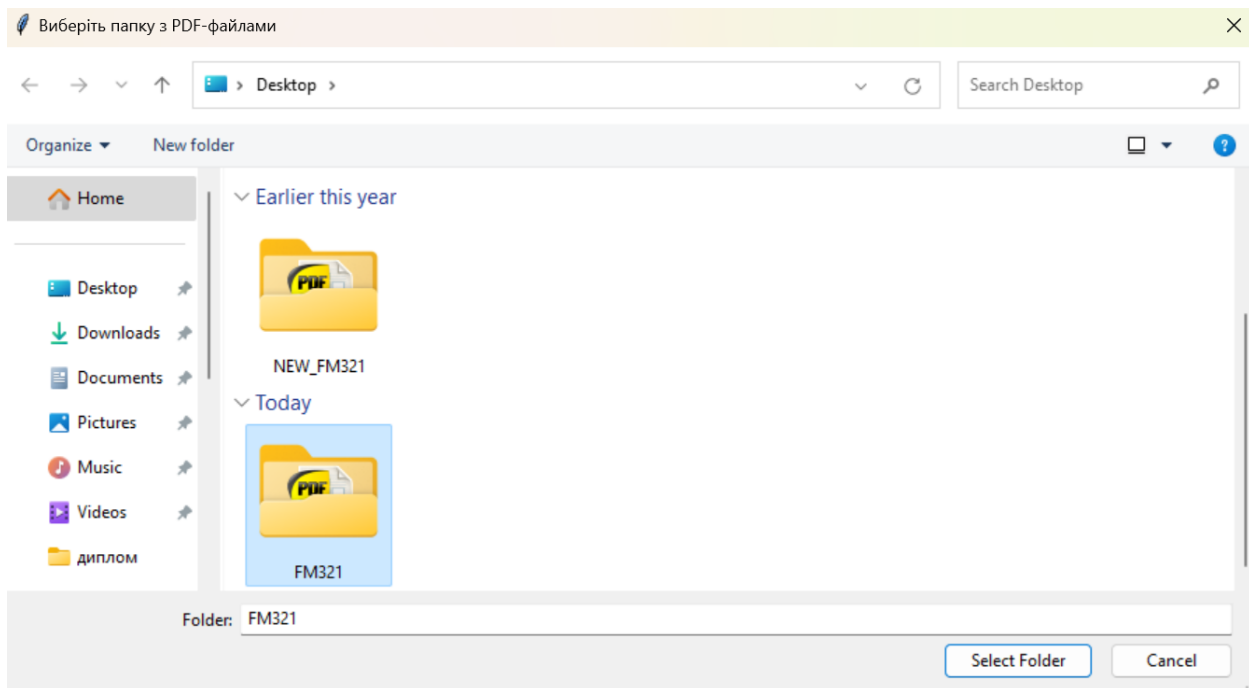


Рис. В2 - Тест 1, вибір папки з PDF-файлами.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\BVA> & C:/Users/BVA/python/python.exe c:/Users/BVA/Desktop/312-all_test.py
Успішно оновлено: fm321-5.pdf
Успішно оновлено: fm321-13.pdf
Успішно оновлено: fm321-21.pdf
Успішно оновлено: fm321-28.pdf
Успішно оновлено: fm321-35.pdf
Успішно оновлено: fm321-42.pdf
Успішно оновлено: fm321-50.pdf
Успішно оновлено: fm321-56.pdf
Успішно оновлено: fm321-63.pdf
Успішно оновлено: fm321-67.pdf
Успішно оновлено: fm321-72.pdf
Успішно оновлено: fm321-77.pdf
Успішно оновлено: fm321-87.pdf
Успішно оновлено: fm321-97.pdf
Успішно оновлено: fm321-108.pdf
Успішно оновлено: fm321-115.pdf
Успішно оновлено: fm321-126.pdf
Успішно оновлено: fm321-134.pdf
Успішно оновлено: fm321-146.pdf
Успішно оновлено: fm321-157.pdf
Успішно оновлено: fm321-161.pdf
Успішно оновлено: fm321-166.pdf

Результат:
Успішно оновлено: 22
Помилка: 0
```

Рис. В3 - Тест 1.

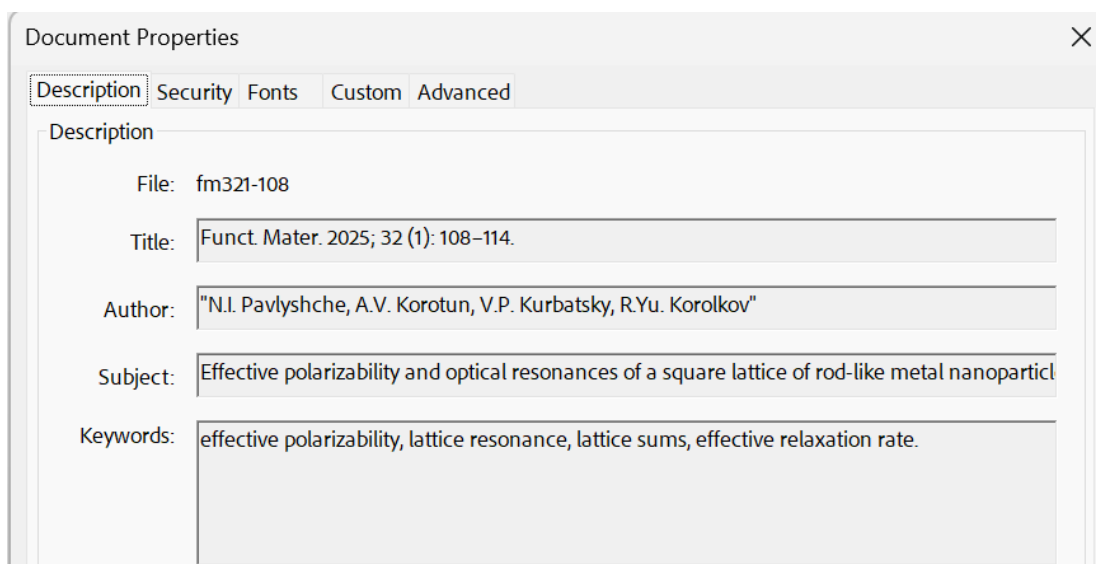


Рис. В4 - Результати тесту 1 одного з оброблених файлів.

ISSN 1027-5495. Functional Materials, 32, No.1 (2025), p. 108-114
 doi:http://dx.doi.org/10.15407/fm32.01.108 © 2025 — STC "Institute for Single Crystals"

Effective polarizability and optical resonances of a square lattice of rod-like metal nanoparticles on a dielectric substrate

*N.I. Pavlyshche¹, A.V. Korotun^{1,2},
V.P. Kurbatsky¹, R.Yu. Korolkov¹*

¹ National University "Zaporizhzhia Polytechnic" 64 University Str.,
Zaporizhzhia, 69063, Ukraine,

² G.V. Kurdyumov Institute for Metal Physics of National Academy of
Sciences of Ukraine,
36 Academician Vernadsky Blvd., Kyiv, 03142, Ukraine

Received September 30, 2024

Resonance properties of the lattice of rod-like nanoparticles are studied in the local field approximation; the characteristics of single nanoparticles are calculated within the framework of the equivalent spheroid approach. According to the results of the calculations, a blue shift of the maxima of the imaginary part of the effective transverse polarizability for the square lattice of nanocylinders with a decrease in their effective aspect ratio was established. The imaginary part of the effective transverse polarization of the lattice of cylinders is four orders of magnitude greater than the corresponding value for a single cylinder. The lattice resonance frequency increases when metals with a higher plasma frequency are used or when the lattice is placed in a medium with a lower dielectric constant. The obtained results indicate a significant influence of the size, shape, and material of rod-like nanoparticles, as well as the surrounding medium, on the frequency dependence of the imaginary part of the effective transverse polarization of the lattice. The possibility of controlling the lattice resonance frequency by changing the effective aspect ratio of the nanoparticles forming the lattice has been demonstrated. The strong interaction between the constituent elements of the lattice leads to an increase by four orders of magnitude in the imaginary part of the effective transverse polarizability and the absorption cross section of the lattice in comparison with a single nanoparticle.

Keywords: effective polarizability, lattice resonance, lattice sums, effective relaxation rate.

Рис. В5 – Додані метадані відповідають змісту статті.

- Тест 2: Відсутність указанного PDF-файлу.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\BVA> & C:/Users/BVA/python/python.exe c:/Users/BVA/Desktop/312-all_test.py
Успішно оновлено: fm321-5.pdf
Успішно оновлено: fm321-13.pdf
Успішно оновлено: fm321-21.pdf
Успішно оновлено: fm321-28.pdf
Успішно оновлено: fm321-35.pdf
Успішно оновлено: fm321-42.pdf
Файл не знайдено: fm321-50.pdf
Файл не знайдено: fm321-56.pdf
Файл не знайдено: fm321-63.pdf
Успішно оновлено: fm321-67.pdf
Успішно оновлено: fm321-72.pdf
Успішно оновлено: fm321-77.pdf
Успішно оновлено: fm321-87.pdf
Успішно оновлено: fm321-97.pdf
Успішно оновлено: fm321-108.pdf
Успішно оновлено: fm321-115.pdf
Успішно оновлено: fm321-126.pdf
Успішно оновлено: fm321-134.pdf
Успішно оновлено: fm321-146.pdf
Успішно оновлено: fm321-157.pdf
Успішно оновлено: fm321-161.pdf
Успішно оновлено: fm321-166.pdf

Результат:
Успішно оновлено: 19
Помилка: 3
```

Рис. В6 - Тест 2.

- Тест 3: Відсутність необхідних полів у CSV.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\BVA> & C:/Users/BVA/python/python.exe c:/Users/BVA/Desktop/312-all_test.py
Помилка читання CSV: No columns to parse from file
```

Рис. В7 - Тест 3.

Виконавець

Студентка групи КУ41 Боровкова Віра



ЛІСТИНГ

```

import os
import re
import pandas as pd
import tkinter as tk
from tkinter import filedialog
from pikepdf import Pdf, Name, String
from lxml import etree

# видалення HTML-тегів
def remove_html_tags(text):
    if pd.isna(text):
        return ""
    return re.sub(r'<.*?>', '', str(text)).strip()

# форматування назви
def extract_title_string(shortname):
    match = re.search(r'(\d+)\s*\((\d+)\):\s*(\d+)-(\d+)',
str(shortname))
    if match:
        volume, issue, start_page, end_page = match.groups()
        return f"Фunct. Mater. 2025; {volume} ({issue}):
{start_page}-{end_page}."
    return "Фunct. Mater."

# обробка авторів: видалення HTML-тегів, індексів організації
тощо
def clean_authors(author_field):
    text = remove_html_tags(author_field).strip()
    text = text.replace("'", '').replace('"', "")
    text = re.sub(r'[\(\[\{\}\?]\d+[\)\]\}\?]', '', text)
    text = re.sub(r'\b\d+\b[.,]?', '', text)
    raw_names = re.split(r'[;,]|\t+|\s{2,}', text)

    cleaned = []
    seen = set()
    for name in raw_names:
        name = name.strip()
        if name and name not in seen:
            cleaned.append(name)
            seen.add(name)

    return ', '.join(cleaned)

# обробка ключових слів
def clean_keywords(keyword_field):
    text = remove_html_tags(keyword_field)
    keywords = re.split(r'[;, \n]', text)

```

```

        cleaned_keywords = [re.sub(r'["\''""]', '', kw.strip()) for
kw in keywords if kw.strip()]
        unique_keywords = []
        for kw in cleaned_keywords:
            if kw not in unique_keywords:
                unique_keywords.append(kw)
        return ', '.join(unique_keywords)

# генерація XMP-метаданих
def set_xmp_metadata(pdf, authors, title, subject, keywords):
    NS = {
        'rdf': 'http://www.w3.org/1999/02/22-rdf-syntax-ns#',
        'dc': 'http://purl.org/dc/elements/1.1/',
        'xmp': 'http://ns.adobe.com/xap/1.0/'
    }

    xmpmeta = etree.Element("{adobe:ns:meta/}xmpmeta",
nsmap={'x': 'adobe:ns:meta/'})
    rdf = etree.SubElement(xmpmeta,
"{http://www.w3.org/1999/02/22-rdf-syntax-ns#}RDF")

    desc = etree.SubElement(rdf,
"{http://www.w3.org/1999/02/22-rdf-syntax-ns#}Description",
nsmap={'dc': NS['dc'], 'xmp':
NS['xmp']})

    # authors
    if authors:
        creator = etree.SubElement(desc,
"{http://purl.org/dc/elements/1.1/}creator")
        seq = etree.SubElement(creator,
"{http://www.w3.org/1999/02/22-rdf-syntax-ns#}Seq")
        for name in authors.split(';'):
            li = etree.SubElement(seq,
"{http://www.w3.org/1999/02/22-rdf-syntax-ns#}li")
            li.text = name.strip()

    # title
    title_el = etree.SubElement(desc,
"{http://purl.org/dc/elements/1.1/}title")
    alt = etree.SubElement(title_el,
"{http://www.w3.org/1999/02/22-rdf-syntax-ns#}Alt")
    li = etree.SubElement(alt, "{http://www.w3.org/1999/02/22-
rdf-syntax-ns#}li")
    li.text = title or "Untitled"

    # subject
    if subject:
        description = etree.SubElement(desc,
"{http://purl.org/dc/elements/1.1/}description")

```

```

        alt = etree.SubElement(description,
"{http://www.w3.org/1999/02/22-rdf-syntax-ns#}Alt")
        li = etree.SubElement(alt,
"{http://www.w3.org/1999/02/22-rdf-syntax-ns#}li")
        li.text = subject

    # keywords
    if keywords:
        subject_el = etree.SubElement(desc,
"{http://purl.org/dc/elements/1.1/}subject")
        bag = etree.SubElement(subject_el,
"{http://www.w3.org/1999/02/22-rdf-syntax-ns#}Bag")
        for kw in keywords.split(','):
            li = etree.SubElement(bag,
"{http://www.w3.org/1999/02/22-rdf-syntax-ns#}li")
            li.text = kw.strip()

    metadata_xml = etree.tostring(xmpmeta, pretty_print=True,
xml_declaration=True, encoding='utf-8')
    pdf.Root.Metadata = pdf.make_stream(metadata_xml)
    pdf.Root.Metadata.Type = Name.Metadata
    pdf.Root.Metadata.Subtype = Name.XML

# додавання метаданих до PDF
def add_metadata(pdf_path, row):
    try:
        with Pdf.open(pdf_path, allow_overwriting_input=True)
as pdf:
            authors = clean_authors(row.get("Authors", ""))
            reviews = remove_html_tags(row.get("Reviews", ""))
            keywords = clean_keywords(row.get("Keywords", ""))
            title = extract_title_string(row.get("Shortname",
""))

            # стандартні метадані
            pdf.docinfo[Name.Title] = String(title)
            pdf.docinfo[Name.Author] = String(authors)
            pdf.docinfo[Name.Subject] = String(reviews)
            pdf.docinfo[Name.Keywords] = String(keywords)
            # XMP-метадані
            set_xmp_metadata(pdf, authors, title, reviews,
keywords)

            pdf.save(pdf_path)
            return True
    except Exception as e:
        print(f"Помилка при обробці {pdf_path}: {str(e)}")
        return False

# головна функція
def main():
    root = tk.Tk()
    root.withdraw()

```

```

csv_path = filedialog.askopenfilename(
    title="Виберіть CSV-файл",
    filetypes=[("CSV файли", "*.csv")]
)
if not csv_path:
    print("CSV-файл не обрано.")
    return
try:
    df = pd.read_csv(csv_path, delimiter="|")
except Exception as e:
    print(f"Помилка читання CSV: {str(e)}")
    return

pdf_folder = filedialog.askdirectory(title="Виберіть папку
з PDF-файлами")
if not pdf_folder:
    print("Папка не обрана.")
    return

files_updated = 0
errors = 0
for _, row in df.iterrows():
    try:
        url_part = str(row['URL']).split('/')[-1]
        pdf_filename = f"fm321-{url_part}.pdf"
        pdf_path = os.path.join(pdf_folder, pdf_filename)
        if os.path.exists(pdf_path):
            if add_metadata(pdf_path, row):
                print(f"Успішно оновлено:
{pdf_filename}")
                files_updated += 1
            else:
                errors += 1
        else:
            print(f"Файл не знайдено: {pdf_filename}")
            errors += 1
    except Exception as e:
        print(f"Критична помилка: {str(e)}")
        errors += 1
print(f"\nРезультат:")
print(f"Успішно оновлено: {files_updated}")
print(f"Помилки: {errors}")

if __name__ == "__main__":
    main()

```