

ПРОГРАМНА РЕАЛІЗАЦІЯ МЕТОДІВ АНАЛІЗУ ТА НОРМОКОНТРОЛЮ ТЕКСТОВИХ ТЕХНІЧНИХ ЗВІТІВ

Д. С. Сидорко, В. С. Глухов

Національний університет “Львівська політехніка”,
кафедра електронних обчислювальних машин
E-mail: DmytroSydorko@gmail.com; Valerii.S.Hlukhov@lpnu.ua

© Сидорко Д. С., Глухов В. С., 2024

У статті представлено розвиток програмної реалізації методів аналізу та нормоконтролю текстових технічних звітів у форматі docx з метою перевірки їх відповідності встановленим вимогам. Запропоноване рішення використовує технології .NET, WPF та DocumentFormat.OpenXml для перевірки коректності форматування сторінок, послідовності стилів, наявності порожніх рядків та відповідності параметрів сторінок стандартам (A4, Letter). Також адміністраторам надано можливість гнучко налаштовувати послідовність стилів за допомогою параметрів “before” і “after”.

Однією з ключових особливостей реалізації є оптимізація розміру програми, яка дозволила зменшити її обсяг її попередньої версії на понад 50 % без втрати функціональності. Усі виявлені програмою невідповідності фіксуються у звітному файлі для подальшого аналізу. Програмне рішення спрощує процес нормоконтролю документації, підвищує якість документів та сприяє стандартизації звітності в організаціях, зменшуючи витрати часу та ресурсів.

Результати роботи демонструють, що автоматизований підхід до нормоконтролю документів, таких як пояснювальні записки до бакалаврських та магістерських кваліфікаційних робіт, здатен значно підвищити якість обробки даних та полегшити управління документацією, сприяючи подальшому розвитку комп’ютерної інженерії та оптимізації робочих процесів.

Ключові слова: Text analysis, Document analysis, Standard control of documents, MS Word, Docx.

Вступ

Освітні, наукові та проектні заклади стикаються з підвищеними вимогами до оформлення документів, що включають не лише їхнє правильне форматування та стилістику, але й необхідність дотримання встановленої структури документа. Невідповідність цим вимогам ускладнює обробку й аналіз інформації та може впливати на ефективність управління документацією. Тому актуальною є розробка рішень, що автоматизують нормоконтроль документів і забезпечують високу якість звітів та технічної документації.

У цій роботі розвинено раніше запропонований підхід до здійснення нормоконтролю звітів та технічної документації шляхом створення програмних інструментів для аналізу та контролю таких документів. Програмне забезпечення, розроблене з використанням .NET, WPF та DocumentFormat.OpenXml, дозволяє виконувати перевірку стилів, відповідність сторінок стандартам (A4, Letter), а також контролювати наявність порожніх рядків і правильність порядку стилів. Ключова

особливість рішення – можливість адміністратора гнучко налаштувати послідовність стилів за допомогою параметрів “before” і “after”.

Оптимізація програми дозволила скоротити розмір попередньої версії більш ніж на 50 %, зберігаючи всі необхідні функції. Це забезпечує швидкість і зручність використання, знижує витрати ресурсів та підвищує ефективність перевірок. Застосування запропонованого підходу сприятиме підвищенню якості та стандартизації документації, що є важливим для подальшого розвитку комп'ютерної інженерії та управління процесами в організаціях.

Огляд літературних джерел

Створення текстових документів, що відповідають вимогам форматування та стилістики, є важливим завданням як для освітнього процесу, так і для проектної діяльності. Недотримання цих вимог може ускладнити аналіз і повторне використання документів. У роботі [1] запропоновано програмний інструмент для контролю за відповідністю документів вимогам. Хоча цей підхід автоматизує частину перевірок, він має багато недоліків, таких як несумісність з документами, які використовувались різними версіями docx і мають велику кількість стилів. Також розміри сторінки і орієнтація не перевіряється. Програма не має можливості встановлювати бажаний порядок дозволених стилів, порожні рядки ігноруються, а також безпосередні зміни стилів у коректних абзацах.

Для розв'язання цієї проблеми в [2] було розроблено модель, яка розглядає абзаци та речення як вузли, спрощуючи аналіз узгодженості тексту. Проте ця модель ще не забезпечує комплексної перевірки стилів у всіх аспектах. Робота [3] фокусується на автоматичній класифікації документів, що дозволяє спростити перевірку різних типів файлів. Однак подібні підходи можуть бути ресурсоемними, що обмежує їх ефективність у великих корпоративних середовищах.

Деякі дослідження зосереджуються на підвищенні точності аналізу за допомогою функцій втрат [4], але вони не враховують специфіку перевірки стилів у технічних документах. Інший підхід [5] представлено у фреймворку ALDOCX, що дозволяє виявляти шкідливі документи за допомогою мета-функцій. Хоча цей підхід добре працює для виявлення загроз, його функціонал у межах стилістичного аналізу залишається обмеженим.

Розробка Text Guide [6] пропонує методи стиснення тексту без втрати інформації, що знижує обчислювальні витрати. Проте застосування цього методу до структурованих документів із складною стилістикою потребує адаптації. Для глибшого розуміння внутрішньої структури XML-документів у [7] проведено порівняння продуктивності систем зберігання даних. Це дослідження підкреслює, що вибір технології для зберігання даних впливає на швидкість доступу до документів та їх обробку.

Безпека документів залишається важливим напрямом досліджень. У [8] проаналізовано способи приховування даних у файлах MS Word і виявлено вразливості у структурі OOXML. Використання блокчейн-технологій для захисту документів [9] підвищує надійність збереження інформації, але потребує значних обчислювальних ресурсів. Нарешті у [10] запропоновано гібридні рішення для захисту даних у корпоративних середовищах, однак їх інтеграція у процеси перевірки стилів залишається складним завданням.

Стилі текстових документів


Для покращення нормоконтролю пояснювальних записок до бакалаврських та магістерських кваліфікаційних робіт на кафедрі електронних обчислювальних машин Національного університету “Львівська політехніка” було розроблено набір стилів для текстового редактора Word (табл. 1). Цей перелік стилів було використано для розробки програмних засобів для нормоконтролю звітів та технічної документації.

Перелік стилів кафедри ЕОМ

Елемент тексту	Приклад елемента тексту з відповідним стилем	Обов'язковий стиль елемента тексту в документі Word (назва стилю повинна починатися префіксом "ЕОМ: ", крім стилів Заголовок та Зміст, а також стилів з рівнозначними назвами іншими мовами) та його параметри
1	2	3
Кількість сторінок бакалаврських та магістерських кваліфікаційних робіт		Не менше 60 повних сторінок, не більше 100 повних сторінок
Параметри сторінки, розмір паперу		A4, 21 x 29,7 см
Поля: верхнє, нижнє, лівє, правє		2 см, 2 см, 2,5 см, 1,5 см,
Колонки		1
Від верхнього колонтитула до тексту		1,25 см
Від нижнього колонтитула до тексту		1,25 см
Автор	Dmytro Sydorko	ЕОМ: Автор Вирівнювання – по центру Times New Roman 14 Малі прописні напівжирний Інтервал перед 0 пт Інтервал після 0 пт Міжрядковий полуторний
Спеціальність	<i>Спеціальність</i>	ЕОМ: Спеціальність Вирівнювання – по центру Times New Roman 14 курсив Інтервал після 0 пт Міжрядковий полуторний
Бакалаврська кваліфікаційна робота	Бакалаврська кваліфікаційна робота	ЕОМ: Центр Вирівнювання – по центру Times New Roman 14 Міжрядковий полуторний
Науковий керівник	<i>Науковий керівник</i>	<i>ЕОМ: Керівник</i> Вирівнювання – ліворуч Times New Roman 14 Інтервал після 16 пт Міжрядковий о-інарний
Львів–2020	Львів–2024	ЕОМ: Центр Вирівнювання – по центру Times New Roman 14 Міжрядковий полуторний

Продовження табл. 1

1	2	3
Зміст	ЗМІСТ	ЕОМ: ЗАГ1 З НОВ. СТОРІНКИ Вирівнювання – по ширині Times New Roman 14 Усі прописні напівжирний Міжрядковий полуторний
Пункти змісту з 1 цифрою або без цифри	Розділ 1	Оглавление 1 Вирівнювання – ліворуч Times New Roman 14 Усі прописні Міжрядковий одинарний
Пункти змісту з 2 цифрами	1.1. Можливі галузі створення конкурентоспроможної продукції	Оглавление 2 Вирівнювання – ліворуч Times New Roman 14 Міжрядковий одинарний Відступ ліворуч 0,5 см
Перелік умовних скорочень	ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	ЕОМ: ЗАГ1 З НОВ. СТОРІНКИ Вирівнювання – по ширині Times New Roman 14 Усі прописні напівжирний Міжрядковий полуторний
Умовні скорочення	Арифметико-логічний пристрій	ЕОМ: Лівий Вирівнювання – ліворуч Times New Roman 14 Міжрядковий полуторний
Вступ	ВСТУП	ЕОМ: ЗАГ1 З НОВ. СТОРІНКИ Вирівнювання – по ширині Times New Roman 14 Усі великі напівжирний Міжрядковий полуторний
Заголовок 1 (з 1 цифрою)	Теоретична частина	Заголовок 1 Times New Roman 14 напівжирний Міжрядковий полуторний Усі прописні Вирівнювання – по центру Нумерація – по лівому краю Рівень нумерації 1
Назва заголовка 1	Аналіз загальних принципів побудови та проектування <<<Назва роботи>>>	ЕОМ: ЗАГ1 без номера Times New Roman 14 напівжирний Міжрядковий полуторний Усі прописні Вирівнювання – по центру
Заголовок 2 (з двома цифрами)	Можливі галузі створення конкурентоспроможної продукції	Заголовок 2 Times New Roman 14 Міжрядковий полуторний Вирівнювання – по ширині Рівень нумерації 2 Відступ 1,25 см Інтервал перед 6 пт Інтервал після 6 пт Положення номеру з лівого краю на 1,25 см Не відривати від наступного

1	2			3
Текст	Бакалаврську кваліфікаційну роботу присвячено ...			ЕОМ: Основний Відступ 1,25 см Times New Roman 14 Міжрядковий полуторний Вирівнювання – по ширині
Номер таблиці	Таблиця 5.1			ЕОМ: Номер таблиці Вирівнювання – праворуч Times New Roman 14 курсив Міжрядковий полуторний Не відривати від наступного Назва для нумерації – Таблиця
Назва таблиці	<i>Характеристики вузлів СП</i>			ЕОМ: Назва таблиці Вирівнювання – по центру Times New Roman 14 напівжирний курсив Міжрядковий полуторний Не відривати від наступного
Таблиця ліва	Data type code	Short notation	Data type description	ЕОМ: Таблиця_лів Times New Roman 12 напівжирний (шапка) Міжрядковий Одинарний Вирівнювання – ліворуч
	'0000'b = 0		Register is free for writing	
Таблиця центральна	Data type code	Short notation	Data type description	ЕОМ: Таблиця_центр Times New Roman 12 напівжирний (шапка) Міжрядковий Одинарний Вирівнювання – по центру
	'0000'b = 0		Register is free for writing	
Номер та назва рисунка	<i>Рис. 5.1. Структурна схема одного з СП</i>			ЕОМ: Назва рисунку Вирівнювання – по центру Times New Roman 12 курсив Міжрядковий полуторний Назва для нумерації – Рис.
Рисунок				ЕОМ: Рисунок Вирівнювання – по центру Times New Roman 12 Міжрядковий полуторний Не відривати від наступного
Висновки	ВИСНОВКИ			ЕОМ: ЗАГІ З НОВ. СТОРІНКИ Вирівнювання – по ширині Times New Roman 14 Усі великі напівжирний Міжрядковий полуторний
Список використаних джерел	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ			ЕОМ: ЗАГІ З НОВ. СТОРІНКИ Вирівнювання – по ширині Times New Roman 14 Усі великі напівжирний Міжрядковий полуторний
Літературні джерела	<i>1. David A. Patterson Software Interface, 2009, 940 p.</i>			ЕОМ: Література Times New Roman 14 Міжрядковий полуторний Назва для нумерації – Література
Гіперпосилання на літературні джерела				Повинні бути обов'язково

Закінчення табл. 1

1	2	3
Гіперпосилання на таблиці		Повинні бути обов'язково
Гіперпосилання на рисунки		Повинні бути обов'язково
Гіперпосилання на Додатки		Повинні бути обов'язково
Не використовувати подвійні пробіли, Використовувати тільки одинарні пробіли		Не використовувати подвійні пробіли, Використовувати тільки одинарні пробіли
Автоматична нумерація рисунків		Повинна бути обов'язково
Автоматична нумерація таблиць		Повинна бути обов'язково
Автоматична нумерація літературних джерел		Повинна бути обов'язково
Не використовувати стилі назви який не починаються з ЕОМ:		Можна використовувати тільки стилі Заголовок 1 Заголовок 2 Зміст 1 Зміст 2 Перелік рисунків
Не повинно бути подвійних пробілів		Не повинно бути подвійних пробілів
Не повинно бути порожніх рядків		Не повинно бути порожніх рядків

Постановка завдання

Завданням цього дослідження є удосконалення програмного забезпечення для підвищення ефективності управління та аналізу текстової документації у форматі docx згідно із встановленими вимогами.

Ця робота зосереджується на оптимізації розміру попередньої версії [1] програми для мінімізації її обсягу без втрати функціональності. Крім того, система забезпечить автоматичну перевірку послідовності стилів у документах і дозволить адміністраторам вносити зміни до цієї послідовності. Програмне забезпечення також має перевіряти відповідність формату сторінок стандартам А4 або Letter, коректність орієнтації сторінок, а також відповідати вимогам щодо розміру полів, колонтитулів та відсутності зайвих порожніх рядків, надаючи детальні звіти про знайдені невідповідності. А також можливість відстеження усіх змін уже присвоєних стилів безпосередньо у тексті.

Результати досліджень

На основі вищепроведених досліджень будують блок схеми алгоритмів ключових перевірок на дотримання норм документа програми.

Алгоритм перевірки параграфів (рис. 1) фокусується на аналізі всіх параграфів, що містяться в основній частині документа, і забезпечує перевірку їхньої коректності відповідно до заданих стилів і структурних вимог.

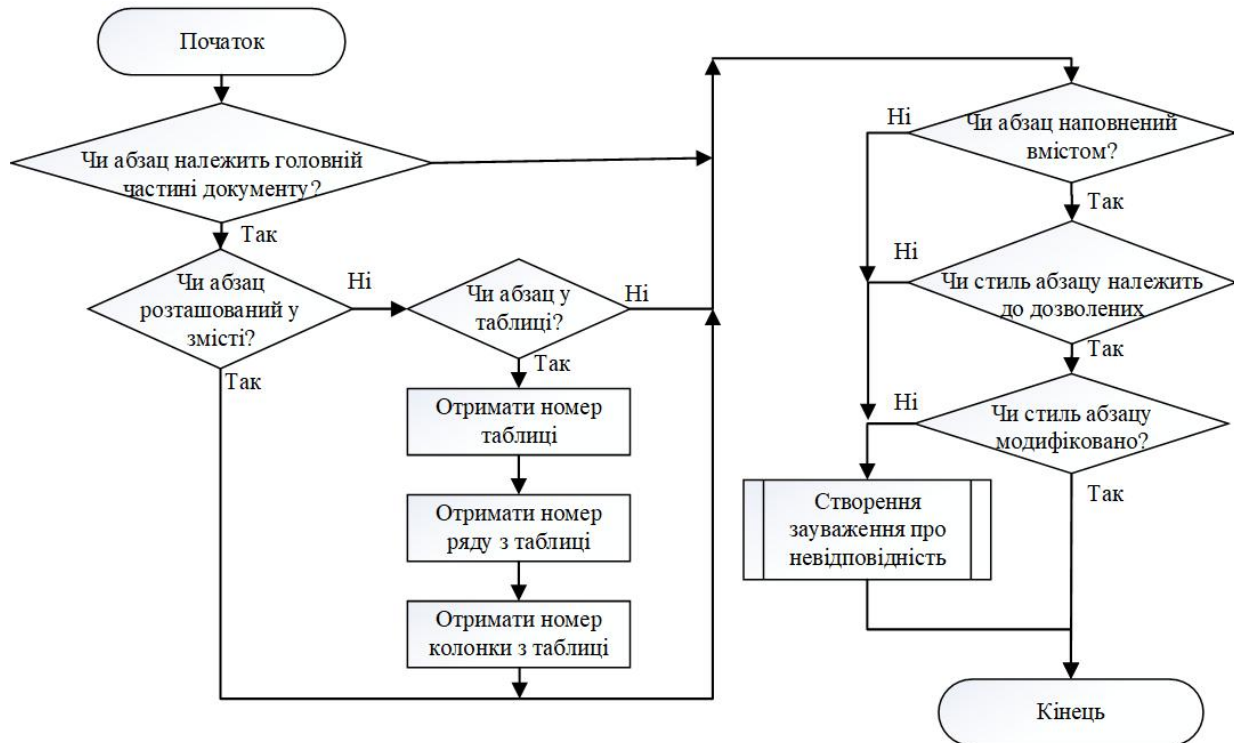


Рис. 1. Алгоритм аналізу головної частини документа (Body)

Алгоритм починає з перевірки наявності параграфів у основній частині документа. Якщо вони відсутні, робота завершується, інакше здійснюється їхній подальший аналіз. Спочатку перевіряється, чи належить параграф до змісту (ТОС); такі абзаци пропускаються, оскільки генеруються автоматично. Якщо це не зміст, алгоритм визначає, чи є параграф частиною таблиці. У разі підтвердження виконується додатковий аналіз для точного позиціонування абзацу в таблиці. Далі перевіряється відповідність стилю. У разі невідповідності або відсутності параграфа генерується звіт про помилку. Якщо ж усі вимоги виконано, алгоритм переходить до наступного параграфа.

Алгоритм (рис. 2) представляє логічну структуру для перевірки коректного порядку стилів у текстових документах, що є важливим етапом аналізу та нормоконтролю технічних звітів. Такий процес дозволяє забезпечити відповідність текстових стилів визначеним стандартам або внутрішнім вимогам компанії, що особливо важливо для документів технічної або наукової спрямованості.

Алгоритм перевірки стилів забезпечує логічну послідовність параграфів у документі. Він аналізує поля “before” та “after” у налаштуваннях стилю, щоб перевірити, чи відповідають сусідні параграфи встановленим правилам. Якщо вказано поле “before”, алгоритм перевіряє, чи попередній параграф має дозволений стиль. Аналогічно, поле “after” визначає, які стилі можуть слідувати після поточного параграфа. Виявлення порушень у послідовності або стилістичних невідповідностей завершується повідомленням про помилку. Якщо всі перевірки пройдено успішно, робота алгоритму завершується, підтверджуючи правильний порядок і відповідність стилів у документі.

Для візуалізації логіки роботи програми перевірки документів, демонструючи різницю між діями користувача та адміністратора, побудована структурна схема (рис. 3). Вона охоплює процеси автентифікації, налаштування параметрів перевірки та аналізу документів, що дозволяють забезпечити коректність і безпомилковість обробки файлів.

Запуск програми можна здійснювати як з правами адміністратора, так і звичайним користувачем. Обдві дають змогу завантажувати документи для перевірки, включати або вимикати генерацію копії та сувору перевірку. Якщо перевірка документа успішна, можна переглянути

результат. Адміністратор, пройшовши авторизацію, відкриває розширені можливості, такі як зміна паролю та налаштування параметрів перевірки текстових документів. Будь-які зміни налаштувань можна зберегти або скасувати. Алгоритм завершується після виконання відповідних дій або виходу з програми.

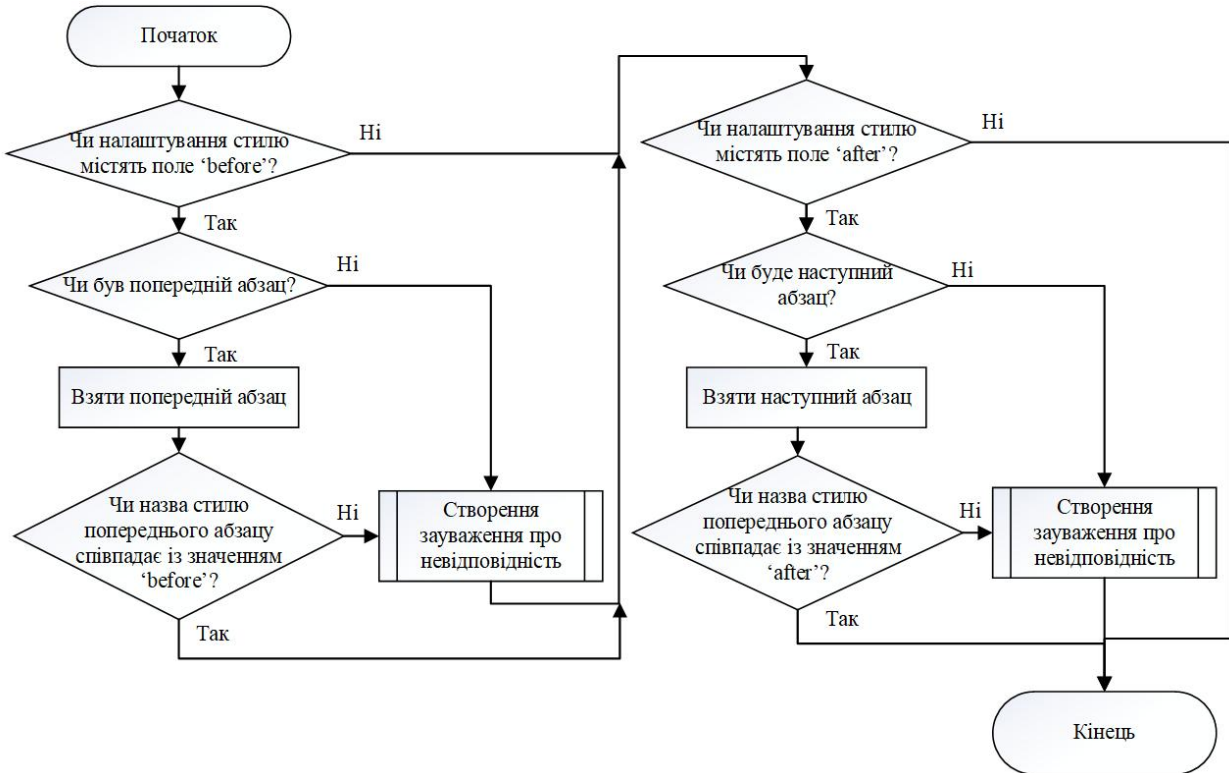


Рис. 2. Алгоритм аналізу коректного порядку стилів

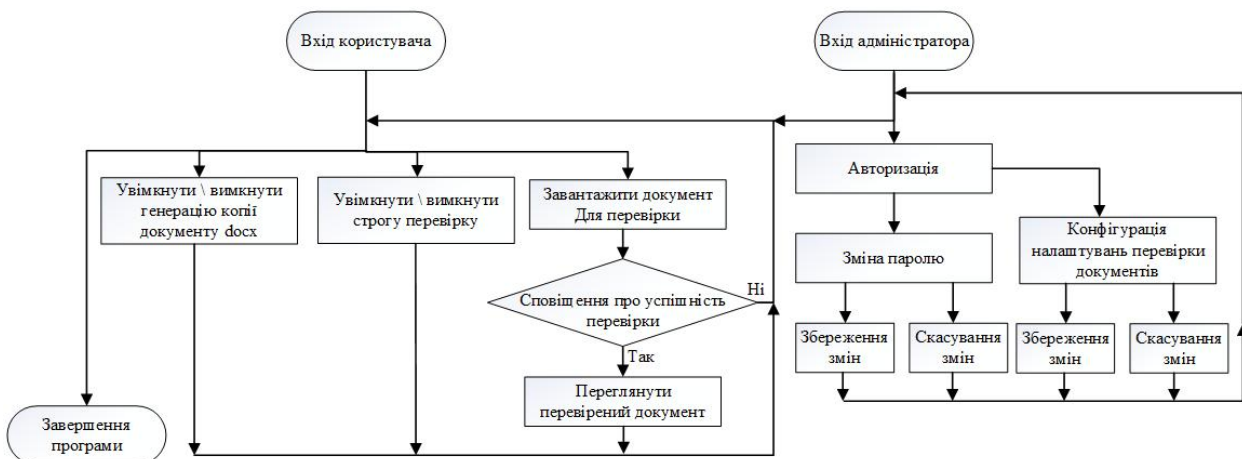


Рис. 3. Структурна схема програми

Реалізація та експериментальна перевірка програми аналізу документів

У новій версії програми інтерфейс зберіг загальну структуру попередньої версії, що забезпечує зручність для постійних користувачів.

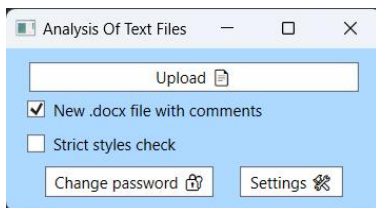


Рис. 4. Вигляд вікна програми для адміністратора

Проте внесено незначні покращення: кнопки зміни паролю та налаштувань отримали більш інтуїтивні назви для кращого розуміння (рис. 4).

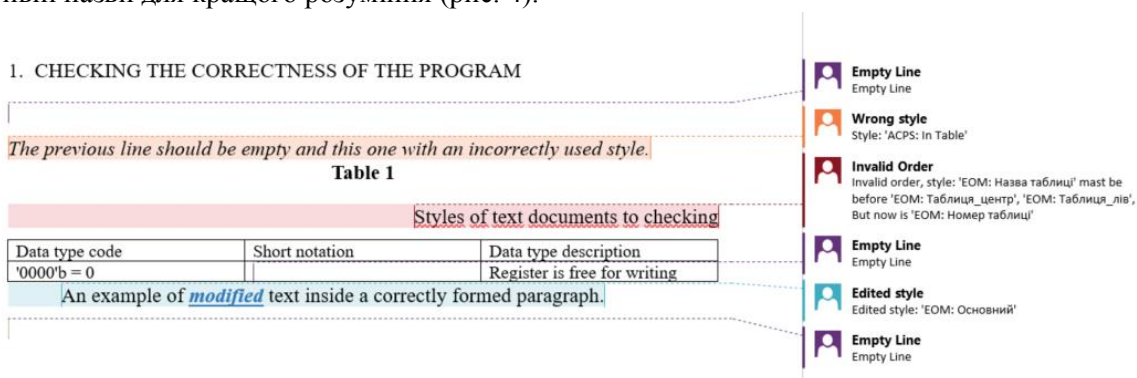


Рис. 5. Висвітлення помилок у файлі ANALISED

Також додано новий чекбокс “Strict styles check,” який активує сувору перевірку стилів, підсвічуючи модифікації конкретних слів або груп. Якщо виявлено відхилення, система генерує попередження з текстом “Edited style” (рис.5).

До функціоналу програми додано перевірку наявності порожніх рядків. Якщо такий рядок знайдено, у звіті з’являється повідомлення з текстом “Empty Line” (рис. 5).

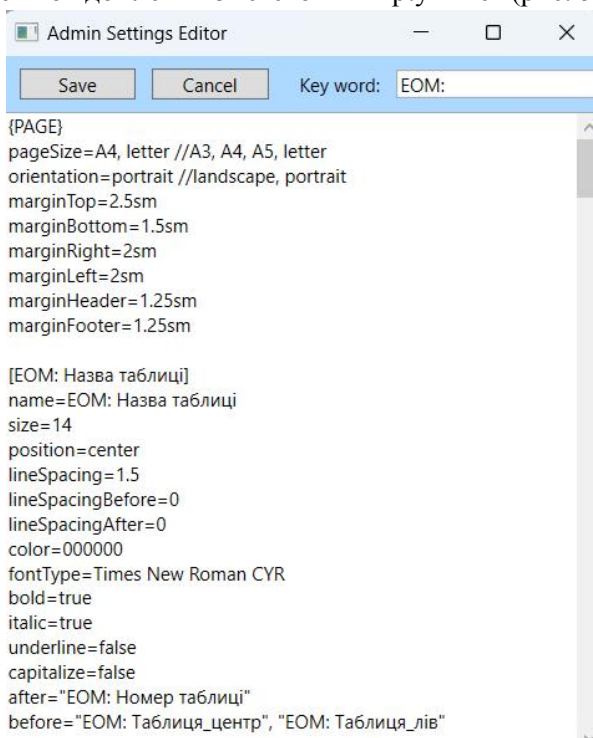


Рис. 6. Налаштування стилів для перевірки

Адміністратор також отримав можливість налаштовувати порядок стилів у документі за допомогою параметрів “before” і “after” (рис. 6). Ці параметри дозволяють визначати кілька стилів для кожної позиції, зазначаючи їх через кому. При виявленні невідповідності з’явиться повідомлення з назвою “Invalid order” (рис. 5).

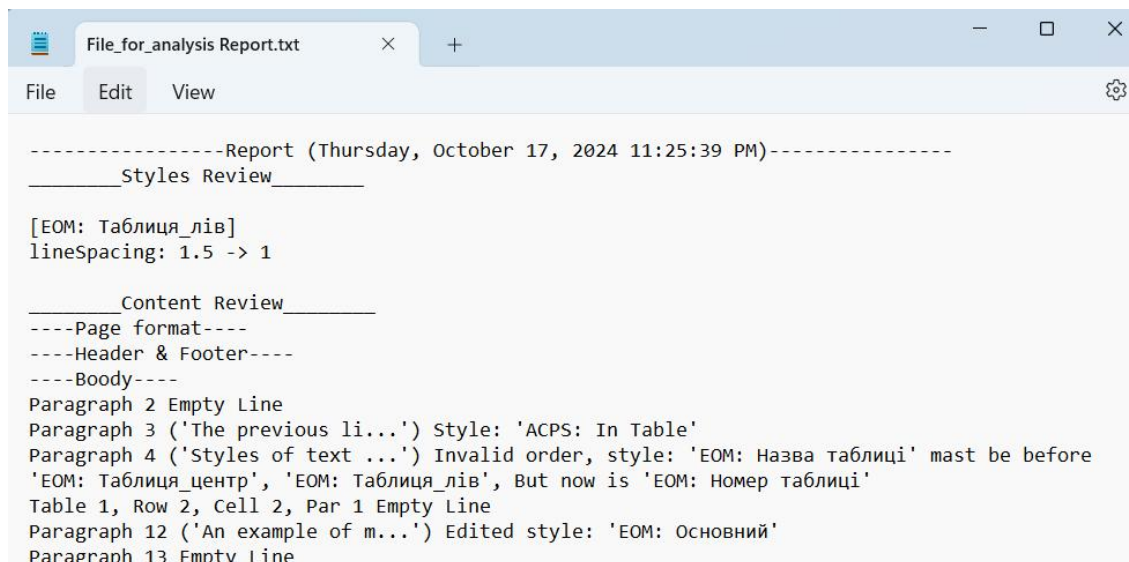


Рис. 7. Вміст файлу Report.txt з виявленими невідповідностями

Адміністратори можуть задати правила перевірки розміру сторінок, орієнтації та відступів як для самої сторінки, так і для колонтитулів (рис. 6). Усі невідповідності фіксуються у звіті “Report.txt” (рис. 7), де детально вказано тип і місце кожної помилки.

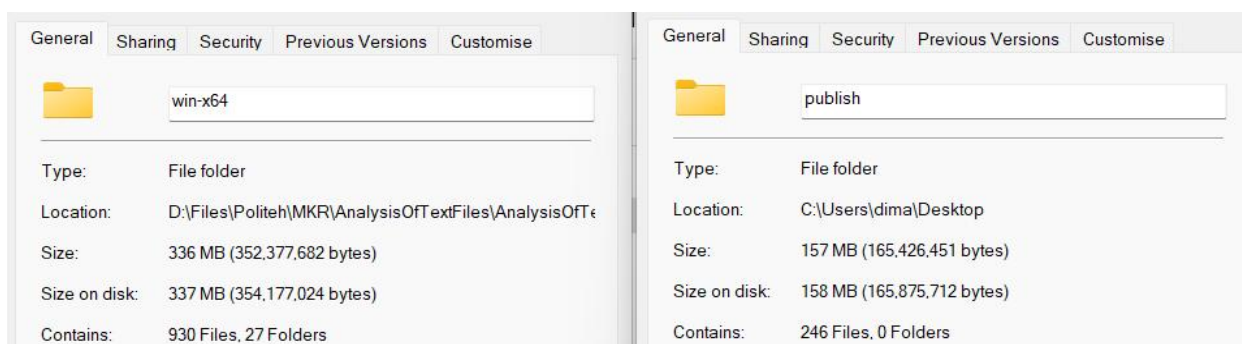


Рис. 8. Порівняння зменшеного розміру вихідного файлу програми з попередньою версією

Одним із пріоритетних завдань стало зменшення обсягу програми без втрати її функціональних можливостей. Завдяки оптимізації коду та використанню стиснення ресурсів вдалося зменшити розмір попередньої версії програми більш ніж на 50 % – від 337 до 157 МБ (рис. 8).

Для перевірки нових функцій було проведено серію тестувань з використанням документів різних форматів і обсягів. Результати показали високу ефективність у виявленні порушень форматування та стилів. Усі виявлені помилки успішно фіксувалися у звіті, а інтерфейс показав зручність і зрозумілість у використанні як для користувачів, так і для адміністраторів.

Програму реалізовано мовою C# з використанням фреймворку .NET і бібліотек DocumentFormat.OpenXml, WPF, в середовищі розробки JetBrains Rider.

Для коректного запуску та роботи програми на комп’ютері з попередньо встановленою системою Windows необхідно встановити лише архів програми, який містить усі необхідні залежності.

Встановлення стилів

У програмі передбачено можливість формування користувачем власних наборів стилів та їх захисту від стороннього втручання. Програму було апробовано для нормоконтролю звітів до бакалаврських, магістерських кваліфікаційних робіт, а також для нормоконтролю статей до журналу ACPS (на відповідність створеного набору стилів журналу ACPS) [11].

Для зміни налаштувань перевірки стилів програму необхідно запустити з правами адміністратора (рис. 9).

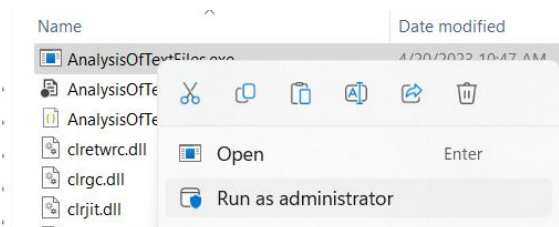


Рис. 9. Запуск програми за правами адміністратора

Інтерфейс відрізняється від звичайного лише наявною кнопкою авторизації (рис. 10), яка відкриває модальне вікно для введення даних авторизації (рис. 11). Пароль за замовчуванням – “admin”.

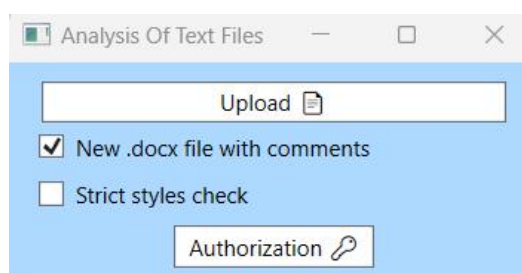


Рис. 10. Інтерфейс програми для неавторизованого адміністратора

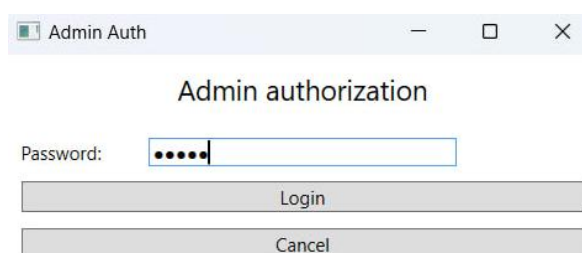


Рис. 11. Модальне вікно авторизації адміністратора

Після успішної авторизації адміністратору надаються дві нові опції: зміна паролю та налаштування критеріїв перевірки стилів (рис. 4). При виборі параметрів налаштувань відкривається модальне вікно, у якому можна коригувати існуючі критерії або додати нові стилі (рис. 6). Доступна також зміна ключового слова (Key Word), яке визначає, з яких символів повинна починатися назва допустимого стилю.

У набірї налаштувань стилів визначає параметри оформлення сторінки та форматування конкретних елементів, зокрема заголовків таблиць, що відповідають вимогам стилістики документа.

Ось детальний опис кожного дозволеного параметра:

Розділ {PAGE} визначає параметри сторінки:

– pageSize: визначає розмір сторінки. Можливі значення: A3, A4, A5, letter.

orientation: визначає орієнтацію сторінки. Можливі варіанти: portrait (портретна), landscape (альбомна).

– marginTop, marginBottom, marginRight, marginLeft: встановлюють верхній, нижній, правий та лівий відступи відповідно (у сантиметрах).

– marginHeader, marginFooter: визначають відступи для верхніх і нижніх колонтитулів (також у сантиметрах).

Усі подальші розділи описують дозвнені стилі і задають параметри для перевірки. Створення конкретного стилю починаються з [Назва стилю], якому можна передати такі поля:

– name: задає назву стилю, яка використовується у документі.

– size: встановлює розмір шрифту.

– position: вирівнювання тексту, може бути “left”, “right”, “center”.

– lineSpacing: міжрядковий інтервал.

– lineSpacingBefore, lineSpacingAfter: інтервал перед і після тексту.

– color: колір тексту в форматі HEX.

– fontType: шрифт, який слід використовувати.

– bold: визначає, чи текст буде жирним.

– italic: визначає, чи буде текст курсивним.

– underline: вказує, чи буде текст підкресленим.

– capitalize: контролює, чи слід перетворювати перші літери слів на великі.

Залежності стилю:

– after: визначає, який елемент, або кілька повинні йти після цього стилю. Зазначаються як “Назва стилю” і через кому “,” можна перелічувати усі дозвнені.

– before: визначає, які елементи повинні передувати цьому стилю, усі налаштування аналогічно параметру “after”.

Якщо після редагування вибрати опцію “Cancel”, внесені зміни не будуть збережені, а вікно закриється. При виборі Save відобразиться статус операції (рис. 12), і якщо не буде виявлено помилок, налаштування збережуться у файлі styleSettings.ini.

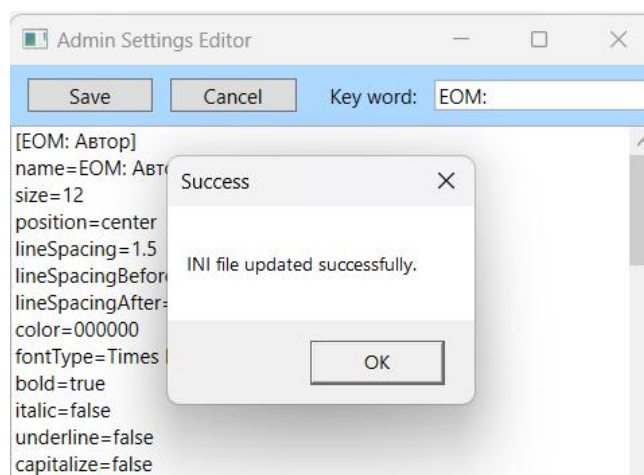


Рис. 12. Повідомлення про успішне збереження стилів

Часові характеристики програми

Результати випробувань продемонстрували високу швидкість роботи програми при аналізі файлів, при тестуванні на різних типах документів із різними обсягами. Зокрема, для документів обсягом 100 сторінок або 5 МБ середній час обробки становив 1,5 секунди (рис. 13). Цей показник свідчить про високу швидкодію програми та демонструє її ефективність порівняно з традиційною ручною перевіркою, яка могла займати значно більше часу.

Для детальнішого тестування було використано кілька наборів документів:

- Звичайний звіт (434 КБ) – проаналізовано за 1,15 секунди.
- Лабораторна робота (1,82 МБ) – проаналізовано за 1,28 секунди.
- Документ БКР (4,78 МБ, 100 сторінок) – обробка зайняла 1,5 секунди.

Ці результати підтверджують, що незалежно від обсягів інформації у документі, програма підтримує стабільну швидкість.

Окрім цього, було перевірено ефективність виявлення стилістичних помилок. Програма продемонструвала високу точність, виявивши 100 % помилок, які були спеціально додані в тестові документи. Це доводить, що автоматизація не лише скорочує час перевірки, але й зменшує ризик пропуску помилок, що часто трапляється під час ручної перевірки.

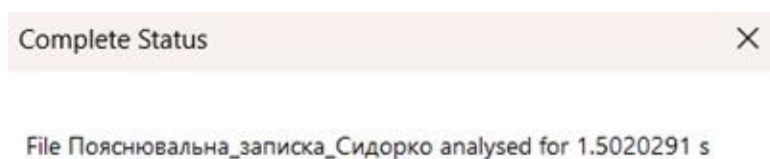


Рис. 13. Часозатратність при аналізі середньостатистичної записки до бакалаврської кваліфікаційної роботи

Висновки

У статті представлено розвиток програмної реалізації методів аналізу та нормоконтролю текстових технічних звітів у форматі docx з метою перевірки їх відповідності встановленим вимогам. Запропоноване рішення використовує технології .NET, WPF та DocumentFormat.OpenXml для перевірки коректності форматування сторінок, послідовності стилів, наявності порожніх рядків та відповідності параметрів сторінок стандартам (A4, Letter). Також адміністраторам надано можливість гнучко налаштувати послідовність стилів за допомогою параметрів “before” і “after”.

Однією з ключових особливостей реалізації є оптимізація розміру програми, яка дозволила зменшити її обсяг її попередньої версії на понад 50 % без втрати функціональності. Усі виявлені програмою невідповідності фіксуються у звітному файлі для подальшого аналізу. Програмне рішення спрощує процес нормоконтролю документації, підвищує якість документів та сприяє стандартизації звітності в організаціях, зменшуючи витрати часу та ресурсів.

Результати роботи демонструють, що автоматизований підхід до нормоконтролю документів, таких як пояснювальні записки до бакалаврських та магістерських кваліфікаційних робіт, здатен значно підвищити якість обробки даних та полегшити управління документацією, сприяючи подальшому розвитку комп’ютерної інженерії та оптимізації робочих процесів.

Завдяки оптимізації коду та використанню стиснення ресурсів вдалося зменшити розмір попередньої версії програми більш ніж на 50 % – від 337 до 157 МБ.

Досягнуто таких часових характеристик програми, як аналіз документів обсягом до 100 сторінок або 5 МБ у середньому за 1,5 секунди, з виявленням 100 % помилок.

Список літератури

1. Hlukhov V. S. and Sydorko D. S. "Algorithms and software for verification of scientific and technical text documents" 2023 *Applied Aspects of Information Technology* 2023; Vol. 6 No. 3:304–317, doi: 10.15276/aait.06.2023.21.
2. Jung D., Kim M. and Cho Y.-S. "Detecting Documents With Inconsistent Context," in *IEEE Access*, vol. 10, pp. 98970–98980, 2022, doi: 10.1109/ACCESS.2022.3204151.
3. Eken S., Menhour H. and Köksal K. "DoCA: A Content-Based Automatic Classification System Over Digital Documents," in *IEEE Access*, vol. 7, p. 97996–98004, 2019, doi: 10.1109/ACCESS.2019.2930339.
4. Nguyen K., Nguyen A., Vo N. D. and Nguyen T. V. "Vietnamese Document Analysis: Dataset, Method and Benchmark Suite," in *IEEE Access*, vol. 10, p. 108046–108066, 2022, doi: 10.1109/ACCESS.2022.3211069.

5. Lee H. and Lee H.-W. "Hidden Message Detection in MS-Word File by Analyzing Abnormal File Structure," 2020 International Conference on Green and Human Information Technology (ICGHIT), Hanoi, Vietnam, 2020, p. 54–57, doi: 10.1109/ICGHIT49656.2020.00021.

6. Xu W., Xu Y., Huo G., Yang Y. and Jin Y. "Optimized Dual-mode Security Encryption Chip Design Based on Hash Algorithm," 2022 IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT), Indore, India, 2022, p. 566–570, doi: 10.1109/CSNT54456.2022.9787655.

7. Kulkarni R. N., Ganesh C., D. K. B K, H. B and Reddy A. P. "Novel Approach to Detect Plagiarism in the Document," 2023 International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), Ballar, India, 2023, p. 1–6, doi: 10.1109/ICDCECE57866.2023.10150442.

8. Chang J. and Fanguy M. "Collab_doc_maker: An Automatic Google-Doc-making Tool," 2021 16th International Conference on Computer Science & Education (ICCSE), Lancaster, United Kingdom, 2021, p. 806–809, doi: 10.1109/ICCSE51940.2021.9569570.

9. Li B., Chen Y. and Zeng L. "Kenet: Knowledge-Enhanced DOC-Label Attention Network for Multi-Label Text Classification," ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Korea, Republic of, 2024, p. 11961–11965, doi: 10.1109/ICASSP48485.2024.10447643.

10. Nissim N., Cohen A. and Elovici Y. "ALDOCX: Detection of Unknown Malicious Microsoft Office Documents Using Designated Active Learning Methods Based on New Structural Feature Extraction Methodology," in IEEE Transactions on Information Forensics and Security, vol. 12, no. 3, p. 631–646, March 2019, doi: 10.1109/TIFS.2016.2631905.

11. Academic Journals and Conferences. *Advances in Cyber-Physical Systems*. Available at: <https://science.lpnu.ua/acps> (Accessed: 18/10/2024)

PROGRAM IMPLEMENTATION OF METHODS FOR ANALYSIS AND VERIFICATION OF TECHNICAL REPORTS

D. Sydorko, V. Glukhov

Lviv Polytechnic National University,
Computer Engineering Department,
Software Department

E-mail: DmytroSydorko@gmail.com; Valerii.S.Hlukhov@lpnu.ua

© Sydorko D., Glukhov V., 2024

The paper provides a solution to the problem of software implementation of methods of analysis and standard control of textual technical reports in docx format to check their compliance with given standards. The proposed solution uses .NET, WPF and DocumentFormat.OpenXml technologies are used to check the correctness of page formatting, the consistency of styles, the presses, and compliance of page parameters with standards (A4, Letter). Administrators are also allowed to flexibly configure the sequence of styles using the “before” and “after” parameters.

One of the key features of the implementation is the optimization of the size of the program, which allowed it to reduce its size by more than 50 % without losing functionality, which ensures high productivity and efficiency. All detected inconsistencies are recorded in a report file for further analysis. The software solution simplifies the process of document verification, improves the quality of documents, and contributes to the standardization of reporting in organizations, reducing the time and resources required.

The results of the work demonstrate that an automated approach to document verification can significantly improve the quality of data processing and facilitate documentation management, contributing to the further development of computer engineering and optimization of work processes.

Keywords: Text analysis, Document analysis, Standard control of documents, MS Word, Docx.