Тема курсової роботи:

Розробка інформаційно-довідкової системи про графік прийому лікарів у поліклініці міста

Виконав:

Студен группи К23-3

Пергун Антон Олександрович

Керівник: Булгакова О. Ф.

Опис проекту

Проект полягає у розробці веб-додатку, який надасть можливість користувачам отримати інформацію про графік прийому лікарів у поліклініці міста. Додаток буде забезпечувати зручний доступ до актуальної інформації про розклад роботи лікарів, допомагаючи пацієнтам ефективно організувати свій візит до поліклініки.

Формулювання задачі

Розробити програму для управління розкладом лікарів у клініці, яка дозволяє зберігати, завантажувати та обробляти інформацію про прийом лікарів. Програма повинна надавати можливість збереження даних у форматі XML, фільтрації та сортування записів, а також виконання різних обчислювальних операцій, таких як перевірка робочого часу кабінету та підрахунок лікарів за певними критеріями.

Завдання

- Вивести інформацію за усіма записами із зазначенням тривалості прийому
- Ввести значення часу і дня тижня прийому; визначити, чи приймає кабінет травматолога в цей час
- Підрахувати кількість лікарів, які починають прийом в п'ятницю в другій половині дня (після 12:00) і вивести відомості про них
- Вивести відомості про терапевта, що останнім завершує сьогодні прийом
- Підготувати інформацію про всіх співробітників, що починають прийом завтра в першій половині дня (до 14:00) і закінчують в другій (після 14:00). Результат надати у вигляді файлу

Застосовані алгоритми:

Алгоритм збереження в XML-форматі: Для забезпечення можливості збереження даних у форматі XML використовується алгоритм, який конвертує інформацію про лікарів у відповідний XML-документ, забезпечуючи зручне зберігання та обмін даними.

Алгоритм фільтрації лікарів: Для фільтрації лікарів, які починають прийом завтра в першій половині дня (до 14:00) і закінчують у другій (після 14:00), використовується алгоритм, який перевіряє дні прийому та часи початку і закінчення прийому, забезпечуючи відбір відповідних записів.

Алгоритм сортування за іменами: Для сортування списку лікарів за їхніми іменами перед збереженням у файл використовується алгоритм сортування, який впорядковує лікарів в алфавітному порядку, забезпечуючи зручний доступ до інформації та можливість аналізу.

Алгоритм двійкового пошуку: використовується для швидкого знаходження елемента в відсортованому масиві або списку. Він працює, розділяючи масив на дві половини і порівнюючи середній елемент з шуканим значенням. На основі цього порівняння, алгоритм вирішує, в якій половині слід продовжувати пошук.

Алгоритм QuickSort: використовується для сортування списку лікарів за часом прийому. Цей алгоритм розділяє список на менші підсписки, сортує їх окремо, а потім об'єднує для отримання остаточного відсортованого списку. Це забезпечує ефективне сортування та швидкий доступ до впорядкованої інформації, що полегшує аналіз даних про прийоми лікарів.

Технології, засоби і методи вирішення задачі:

Використання DataGridView для відображення даних:

Для зручного відображення та обробки даних про лікарів використовується компонент DataGridView, що дозволяє легко додавати, видаляти та редагувати записи.

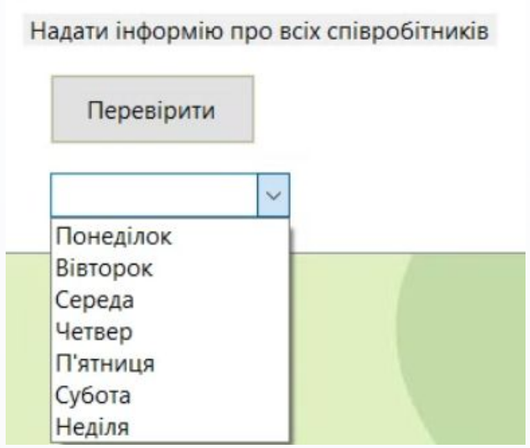
Розробка функції збереження та завантаження даних даних у XML-форматі:

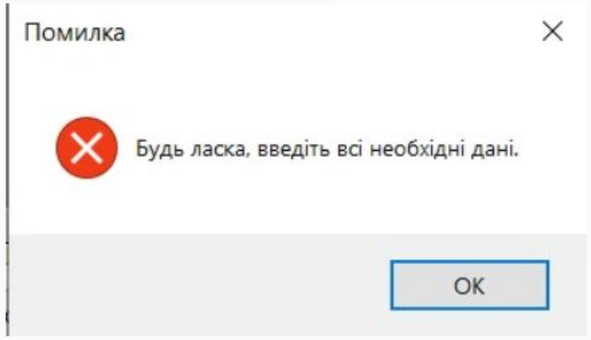
Для забезпечення можливості збереження даних у форматі XML було розроблено алгоритм, який конвертує інформацію про лікарів у відповідний XML-документ. Це дозволяє легко зберігати та обмінюватися даними між різними системами.

Для забезпечення можливості завантаження даних з XML-файлів розроблено функцію, яка зчитує XML-документ і додає інформацію про лікарів у таблицю DataGridView.

Технології, засоби і методи вирішення задачі:

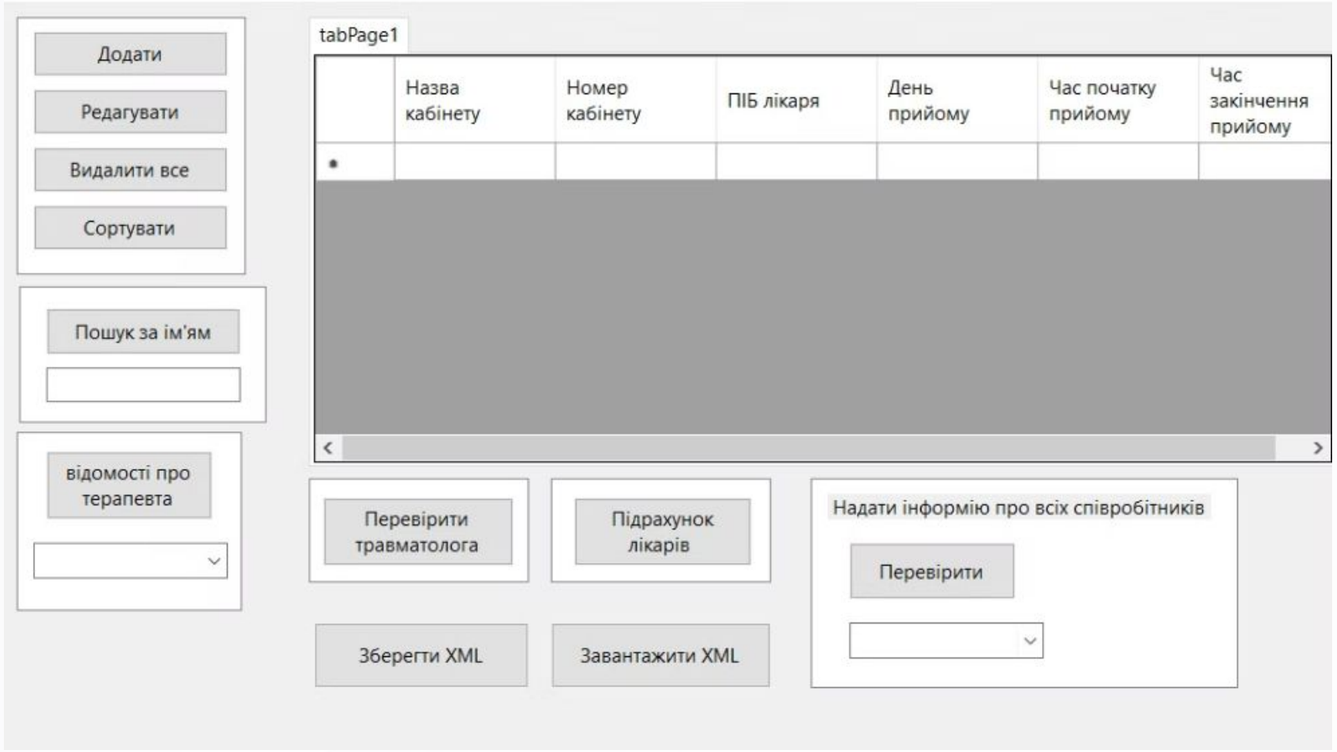
Сповіщення про неправильний формат часу та відсутність потрібного лікаря





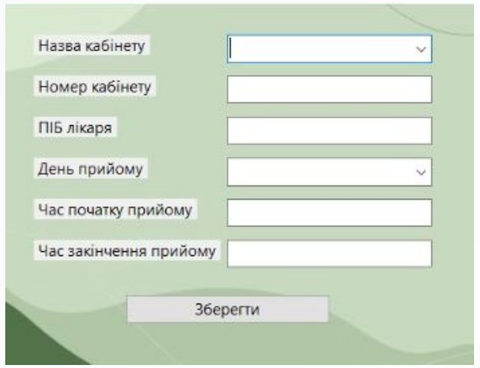
Проектування інтерфейсу

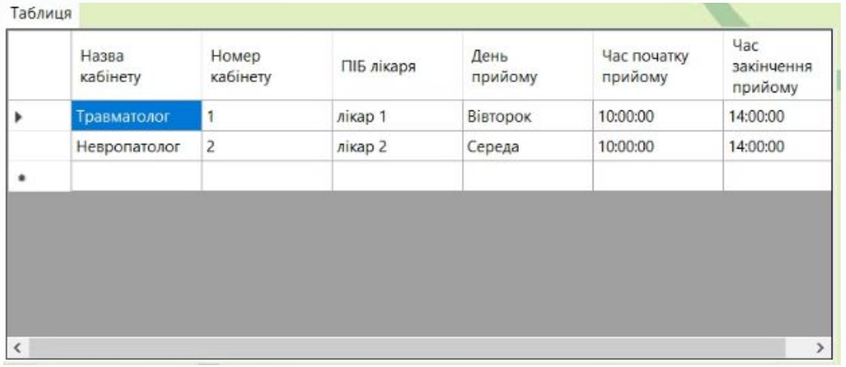
Інтуїтивно зрозумілий та легкий у використанні інтерфейс системи робить її доступною для людей різного віку та рівня комп'ютерної грамотності.



Приклад використання програми:

- Додавання даних до таблиці.

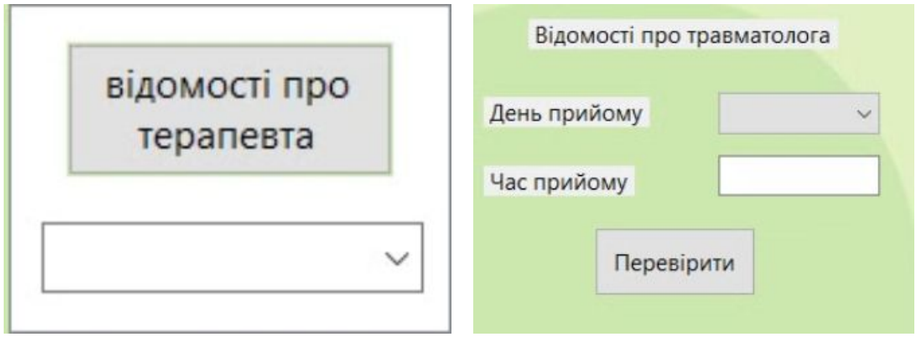


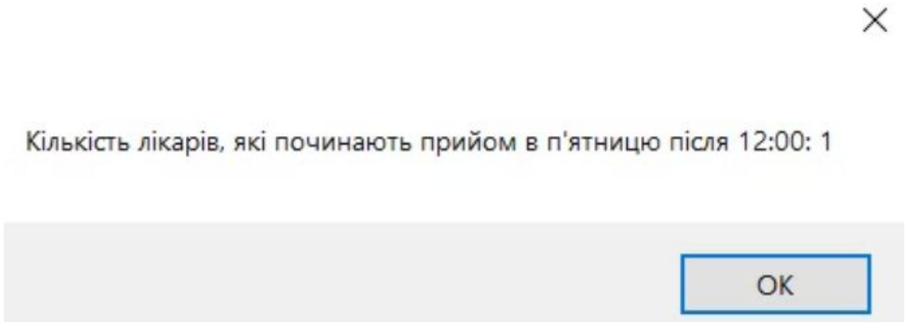


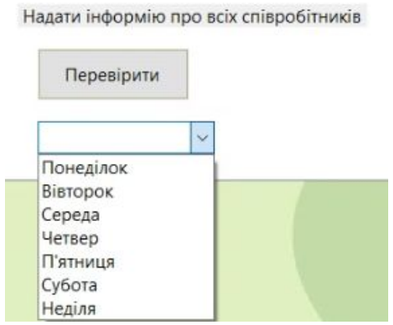
Приклад використання програми:

Перевірка роботи кабінету в зазначений час.

- Підрахунок кількості лікарів, що починають прийом у п'ятницю після 12:00.
- Отримання інформації про останнього терапевта, який завершує прийом сьогодні.
- Генерація звіту про лікарів, що починають прийом завтра в першій половині дня і закінчують в другій.







Висновки:

- Успішна реалізація завдань:
    - Програма коректно виконує всі поставлені задачі.
    - Зручний інтерфейс та простота у використанні.
    - Гнучкість у налаштуванні та масштабованість.
- Продуктивність та ефективність:
    - Швидке оброблення даних.
    - Висока продуктивність при роботі з великими обсягами даних.

Пропозиції щодо вдосконалення:

- Додаткові функціональні можливості:
    - Розширення функцій для аналізу даних.
    - Інтеграція з іншими системами для обміну даними.
- Покращення інтерфейсу користувача:
    - Додавання більш інтуїтивних елементів управління.
    - Підвищення доступності для користувачів з особливими потребами.
- Оптимізація продуктивності:
    - Подальша оптимізація алгоритмів для підвищення швидкодії.
    - Зниження використання ресурсів.
- Безпека та захист даних: