

ВИПУСКНИЙ КВАЛІФІКАЦІЙНИЙ ПРОЄКТ

на тему:

«Програмний модуль парсингу сайту Answer.ua»

Студента 4 курсу, 7 групи,
спеціальності 121 «Інженерія
програмного забезпечення»
освітньої програми «Інженерія
програмного забезпечення»

Прокопчука
Владислава Ігоровича

підпис студента

Науковий керівник
кандидат економічних наук,
доцент кафедри інженерії
програмного забезпечення та
кібербезпеки

Тищенко Дмитро
Олександрович

підпис керівника

Гарант освітньої програми
кандидат технічних наук,
доцент кафедри інженерії
програмного забезпечення та
кібербезпеки

Рзаєва Світлана
Леонідівна

підпис гаранта

Державний торговельно-економічний університет

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення та кібербезпеки

Освітній ступінь бакалавр

Спеціальність 121 «Інженерія програмного забезпечення»

Затверджую

Зав. кафедри інженерії програмного забезпечення та кібербезпеки

Криворучко О. В.

«14» листопада 2022 р.

Завдання

на випускний кваліфікаційний проєкт студентів

Прокопчука Владислава Ігоровича

(прізвище, ім'я, по батькові)

1. Тема випускного кваліфікаційного проєкту «Програмний модуль парсингу сайту Answer.ua»

Затверджена наказом ректора від «6» грудня 2022 р. № 3288

2. Строк здачі студентом закінченого проєкту 5 червня 2023

3. Цільова установка та вихідні дані до проєкту

Мета проєкту - розробка програмного модуля, який дозволить автоматично отримувати та обробляти інформацію про товари та ціни на сайті Answer.ua.

Об'єкт дослідження - сайт Answer.ua.

Предмет дослідження - розробка програмного модуля парсингу для сайту Answer.ua.

4. Консультанти проєкту із зазначенням розділів, які консультують:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв

5. Зміст випускного кваліфікаційного проєкту (перелік питань за кожним розділом)

ВСТУП

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТА ДОСЛІДЖЕННЯ І ОПИС ЙОГО

ОСНОВНИХ ПАРАМЕТРІВ ТА ХАРАКТЕРИСТИК

1.1. Аналіз поточного стану парсингу сайту

1.2. Розробка та впровадження програмного модулю парсингу сайту

1.3. Технічне завдання

1.4. Висновки до розділу 1

РОЗДІЛ 2. ПРОЄКТУВАННЯ ПРОГРАМНОГО МОДУЛЮ

ПАРСИНГУ САЙТУ ANSWEAR.UA

2.1. Чинники, що визначають хід і результати роботи програмного модулю сайту Answer.ua

2.2. Інструменти розробки програмного модулю парсингу сайту Answer.ua

2.3. Висновки до розділу 2

РОЗДІЛ 3. РОЗРОБКА ТА ВПРОВАДЖЕННЯ ПРОГРАМНОГО

МОДУЛЯ ПАРСИНГУ САЙТУ ANSWEAR.UA

3.1. Функціональні характеристики розробленого програмного модулю парсингу сайту Answer.ua

3.2. Особливості впровадження програмного модулю парсингу сайту Answer.ua

3.3. Висновки до розділу 3

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

ДОДАТКИ

6. Календарний план виконання проєкту

№ по р.	Назва етапів випускного кваліфікаційного проєкту	Строк виконання етапів проєкту	
		за планом	фактично
1	2	3	4
1.	<i>Вибір теми випускного кваліфікаційного проєкту</i>	21.09.2022	21.09.2022
2.	<i>Розробка та затвердження завдання на проєкт</i>	14.11.2022	14.11.2022
3.	<i>Вступ та перелік літературних джерел</i>	23.12.2022	23.12.2022
4.	<i>Розділ 1. Аналіз предмета дослідження і опис його основних параметрів та характеристик</i>	27.01.2023	27.01.2023
5.	<i>Розділ 2. Проєктування програмного модулю парсингу сайту Answear.ua</i>	03.03.2023	03.03.2023
6.	<i>Розділ 3. Розробка та впровадження програмного модулю парсингу сайту Answear.ua</i>	14.04.2023	14.04.2023
7.	<i>Висновки</i>	28.04.2023	28.04.2023
8.	<i>Здача випускного кваліфікаційного проєкту на кафедрі (перша перевірка)</i>	17.05.2023	17.05.2023
9.	<i>Підготовка автореферату та презентації доповіді</i>	26.05.2023	26.05.2023
10.	<i>Попередній захист випускного кваліфікаційного проєкту</i>	29.05.2023 – 02.06.2023	
11.	<i>Зовнішнє рецензування випускного кваліфікаційного проєкту</i>	05.06.2023	05.06.2023
12.	<i>Здача прошого випускного кваліфікаційного проєкту на кафедрі</i>	05.06.2023	05.06.2023
13.	<i>Публічний захист випускного кваліфікаційного проєкту</i>		

7. Дата видачі завдання «14» листопада 2022 р.

8. Науковий керівник випускного кваліфікаційного проєкту

Тищенко Д.О.

(прізвище, ініціали, підпис)

9. Гарант освітньої програми

Рзаєва С.Л.

(прізвище, ініціали, підпис)

10. Завдання прийняв до виконання студент

Прокопчук В.І.

(прізвище, ініціали, підпис)

АНОТАЦІЯ

Відповідно до мети дослідження робота присвячена розробці додатку для парсингу сайту Answer.ua

Програмний модуль був створений за допомогою мови програмування C# на платформі WPF, з базою даних MS SQL з Entity Framework.

Розробка серверної частини виконана у середовищі Visual Studio.

У першому розділі будуть розглянуті проблеми, які можуть бути присутні при спробі парсингу даних сайту, а також етапи розробки та впровадження парсеру. У другому розділі будуть обгорені фактори, які впливають на точність, швидкість або чинники, які можуть якимось чином завадити його стабільній роботі. Також будуть наведені приклади вспоміжних інструментів, які можуть допомогти при створенні модулю парсингу. В третьому розділі буде розказано про можливості та характеристики мого програмного модулю.

ABSTRACT

According to the purpose of the study, the work is devoted to the development of an application for parsing the Answer.ua site.

The software module was created using the C# programming language on the WPF platform, with an MS SQL database with Entity Framework.

The development of the server part was carried out in the Visual Studio environment.

The first section will consider problems that may be present when trying to parse site data, as well as the stages of parser development and implementation. In the second section, the factors that affect accuracy, speed or factors that can somehow prevent its stable operation will be burned. Examples of auxiliary tools that can help in creating a parsing module will also be given. The third section will tell about the capabilities and characteristics of my software module.

ЗМІСТ

ВСТУП	4
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТА ДОСЛІДЖЕННЯ І ОПИС ЙОГО ОСНОВНИХ ПАРАМЕТРІВ ТА ХАРАКТЕРИСТИК	6
1.1. Аналіз поточного стану парсингу сайту.....	6
1.2. Розробка та впровадження програмного модулю парсингу сайту.....	9
1.3. Технічне завдання.....	11
1.4. Висновки до розділу 1.....	21
РОЗДІЛ 2 ПРОЄКТУВАННЯ ПРОГРАМНОГО МОДУЛЮ ПАРСИНГУ САЙТУ ANSWEAR.UA	23
2.1. Чинники, що визначають хід і результати роботи програмного модулю парсингу сайту Answer.ua.....	23
2.2. Інструменти розробки програмного модулю парсингу сайту Answer.ua.....	28
2.3. Висновки до розділу 2.....	33
РОЗДІЛ 3 РОЗРОБКА ТА ВПРОВАДЖЕННЯ ПРОГРАМНОГО МОДУЛЮ ПАРСИНГУ САЙТУ ANSWEAR.UA	35
3.1. Функціональні характеристики розробленого програмного модулю парсингу сайту Answer.ua.....	35
3.2. Особливості впровадження програмного модулю парсингу сайту Answer.ua.....	38
3.3. Висновки до розділу 3.....	43
ВИСНОВКИ ТА ПРОПОЗИЦІЇ	44
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	47
ДОДАТКИ	49

					<i>ДТЕУ 121 07-21БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	Програмний модуль парсингу сайту Answer.ua <i>Зміст</i>	<i>Стадія</i>	<i>Арку</i>	<i>Аркушів</i>
Зав. каф.	Криворучко О.В.			23.12.22		3	3	48
Керівник	Тищенко Д.О.			23.12.22		Факультет інформаційних технологій 4 курс, 7 група		
Гарант	Рзаєва С.Л.			23.12.22				
Розробив	Прокопчук В.І.			23.12.22				

ВСТУП

Актуальність. У сучасному світі інтернет-торгівля є надзвичайно популярною, а ринок онлайн-магазинів постійно зростає. Проте, для ефективної роботи в цій сфері, необхідно мати достатньо інформації про товари та ціни, що пропонуються на конкретному сайті. У зв'язку з цим, стає дедалі важливішим розробляти програмні засоби, які дозволять автоматично отримувати та обробляти цю інформацію. Один з таких засобів - програмний модуль парсингу

Мета. Метою дипломної роботи є розробка програмного модуля, який дозволить автоматично отримувати та обробляти інформацію про товари та ціни на сайті Answer.ua.

Завдання. Для досягнення поставленої мети були визначені наступні завдання:

- дослідити особливості структури сайту Answer.ua та визначити найбільш ефективні методи отримання необхідної інформації;
- розробити програмний модуль для автоматичного парсингу сайту Answer.ua;
- перевірити розроблений модуль на практиці та оцінити його ефективність.

Об'єкт. Об'єктом дослідження є сайт Answer.ua.

Предмет. Предметом дослідження є розробка програмного модуля парсингу для сайту Answer.ua.

Методи. У процесі дослідження були використані наступні методи: аналіз літератури та інтернет-джерел, аналіз структури сайту Answer.ua,

					<i>ДТЕУ 121 07-21.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	Програмний модуль парсингу сайту Answer.ua <i>Вступ</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Зав. каф.		Криворучко О.В.		23.12.22		<i>В</i>	4	48
Керівник		Тищенко Д.О.		23.12.22		Факультет інформаційних технологій 4 курс, 7 група		
Гарант		Рзаєва С.Л.		23.12.22				
Розробив		Прокопчук В.І.		23.12.22				

методи програмування та тестування програмного забезпечення сайту Answear.ua.

Практичне значення. Розроблений програмний модуль може бути використаний в інтернет-торгівлі для отримання та обробки інформації про товари та ціни на сайті Answear.ua. Це дозволить підприємствам збільшити ефективність своєї роботи та скоротити час на аналіз конкурентів і цін на товари. Крім того, розробка програмного модуля може стати базою для подальших досліджень та розробок в області парсингу інформації з інших сайтів.



						ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			5

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТА ДОСЛІДЖЕННЯ І ОПИС ЙОГО ОСНОВНИХ ПАРАМЕТРІВ ТА ХАРАКТЕРИСТИК

1.1. Аналіз поточного стану парсингу сайту

Парсер, який виконує функцію парсингу, збирає дані зі структурованого або неструктурованого джерела даних.

В нашому часі, йому можна знайти застосування багато де, адже це дуже зручна функція. Використання її можна знайти в різних галузях, в таких як веб-розробка, аналіз даних.

Але використання його не є таким простим, тому що треба дотримуватися певних правил аби не було проблем. Зараз я наведу приклади і розповім про них, більш детально (рис. 1.1).

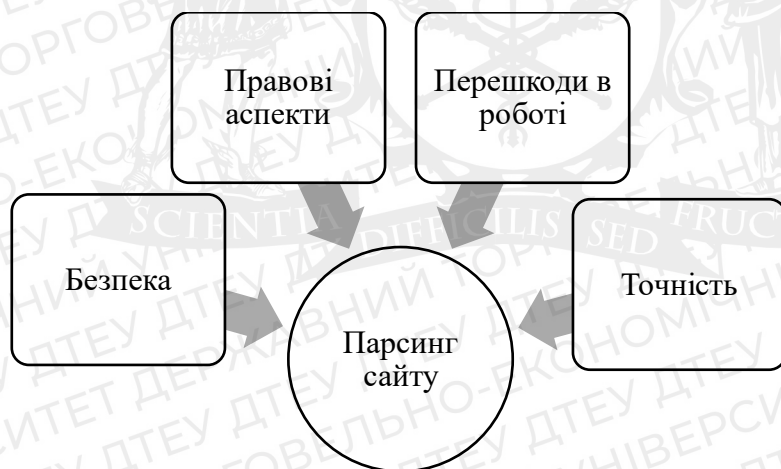


Рис. 1.1. Проблеми парсингу сайту

Джерело: побудовано автором

					ДТЕУ 121 07-21.БР			
Зм.	Аркуш	№ докум.	Підпис	Дата				
Зав. каф.		Криворучко О.В.		27.01.23	Програмний модуль парсингу сайту Answear.ua	Стадія	Аркуш	Аркушів
Керівник		Тищенко Д.О.		27.01.23		РІ	6	48
Гарант		Рзаєва С.Л.		27.01.23		Факультет інформаційних технологій 4 курс, 7 група		
Розробив		Прокопчук В.І.		27.01.23				
					Аналіз предмета дослідження і опис його основних параметрів та характеристик			

Розглянемо дані проблеми за порядком та їх можливі вирішення.

Першій, я хотів би виділити час - це безпеці, адже із зростанням обсягу даних, які отримують завдяки парсингу, ми маємо більший шанс, що вся інформація попаде не в добрі руки. Завдяки цьому, зловмисники можуть використовувати всі дані в своїх цілях, що є дуже прикрим моментом, до якого ніхто не повинен прагнути. Існує багато варіантів, що саме може трапитися з клієнтом сайту, його персональні дані, такі як адреса проживання, ФІБ, пароль та електронна пошта для заходу на сайт, тощо. Це порушує законодавство про захист особистих даних, тим що стає відомим ваш інтернет-трафік та дії в інтернеті.

Але можна завжити деяких заходів безпеки, щоб такого не траплялося. В приклад можу навести, аутентифікацію та шифрування, яке допоможе запобігти отримання персональних даних небажаними персонами. Хочу сказати, що надати максимальну безпеку при парсингу, є одним із найважливіших при розробці програми для парсингу. Законодавство своїми діями має регулювати та уникати таких прецедентів, щоб не було правових проблем.

Наступне – це правові аспекти, які обмежують дії парсингу і зазначають, що є дозволеним в цьому процесі, зазначені в законодавстві. Кожна країна має свої правові наслідки, і беручи в приклад нашу державу, то ми маємо відповідальність за порушення авторських та суміжних прав.

Окрім порушення прав клієнтів сайту, парсинг може порушити права власників сайту. Це також призводить до правових наслідків та несення відповідальності за скоєне. Щоб не мати проблем з авторським правом, можна отримати дозвіл на використання даних від власників. В такому випадку ми уникаємо правових проблем.

						ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			7

Перешкода для парсингу може заповільнити, унеможливити процес, що є наступною проблемою парсингу сайту. Існує автоматизована система, яка визначає хто виконує дії, завдяки виконанню якоїсь дії, наприклад відмітити картинку до якої підходить текст, прожимання галочки чи письмова відповідь на запитання. Це називається CAPTCHA.

Обхід захисту сайту може порушувати законодавство, правові аспекти та авторські права, тому використовувати його потрібно в межах законодавства та етичних норм. Для того, щоб уникнути перешкод в роботі, парсери можуть використовувати різні методи, які дозволяють обійти захист сайту. Наприклад, використання проксі-серверів може допомогти уникнути блокування IP-адресів. Також можна використовувати технології машинного навчання для автоматичного розв'язання CAPTCHA.

Неточність результатів парсингу є наступною проблемою, тому що збираючи дані з різних сайтів, ми маємо різну структуру та формат даних. Розглядаючи один сайт, ми можемо знайти інформацію в такому форматі - назва, фото, технічні характеристики, відгуки, питання, а інший сайт вже має інший формат, наприклад - назва, ціна, опис предмету. Неправильний вибір HTML-тегів для отримання потрібних даних, визначення меж полів в коді програми призводить до неточності парсингу.

Для того, щоб зменшити неточність ми можемо використовувати бібліотеки з вбудованими алгоритмами парсингу тарегулярні вирази, які допомагають зробити вибірку потрібних даних за шаблоном. Це забезпечить нам більшу точність в парсингу. Не буде зайвим нагадати, що треба використовувати парсинг обережно та дотримуючись законодавства та правових аспектів.

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		8

1.2. Розробка та впровадження програмного модулю парсингу сайту

Розробка та впровадження програмного модулю парсингу сайту є складним процесом, який включає в себе наступні етапи (рис. 1.2).

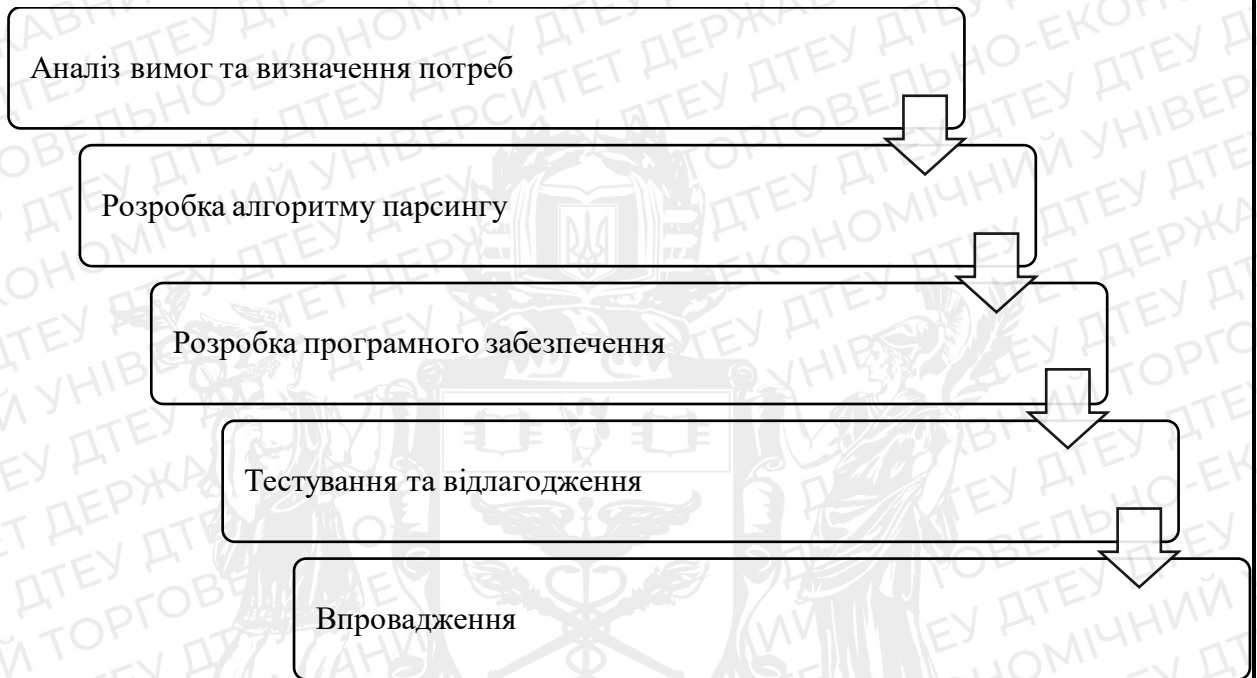


Рис. 1.2. Етапи розробки та впровадження програмного модулю парсингу сайту

Джерело: побудовано автором

На етапі аналізу вимог та визначення потреб, ми повинні вирішити, що ми збираємо з сайту, періодичність та його формат. Збільшити швидкість парсингу та зменшення неточності ми можемо завдяки визначенню точної потреби в наших даних та зменшенню кількості зайвих даних, які буде обробляти парсер.

На етапі розробки алгоритму для парсингу вирішується, яким чином будуть отримуватись та оброблятись дані з сайту. Потрібно визначити спосіб

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		9

доступу до сторінок сайту, з якого будуть збиратись дані, визначити параметри запитів, а також спосіб обробки отриманих даних.

Третім етапом стане етап розробки програмного забезпечення, де потрібно використовувати простий та зрозумілий інтерфейс, врахувати зручність та ефективність, мати гнучкі налаштування та адаптуватись до зміну даних на сайті.

Етап тестування та відлагодження проводиться заради покращення алгоритмів парсингу та точності. Тут проводиться тестування програмного забезпечення на різних сайтах та в різних умовах роботи.

П'ятим етапом є впровадження програмного забезпечення, яке здійснюється після вдалого тестування та відлагодження. Тут потрібно правильно налаштувати параметри парсингу та встановити програму на відповідні пристрої.

Після розробки програмного модулю парсингу сайту, необхідно провести його випробування та відлагодження перед впровадженням. Це дозволить виявити та виправити помилки та неточності в роботі програмного забезпечення.

Для впровадження програмного модулю парсингу сайту необхідно визначити, які сайти будуть парситися та яку інформацію необхідно збирати. Також необхідно визначити час та частоту оновлення даних, щоб забезпечити актуальність інформації.

При впровадженні програмного модулю парсингу сайту необхідно враховувати питання безпеки та правових аспектів. Для запобігання можливих проблем необхідно встановити обмеження на швидкість парсингу, щоб уникнути блокування IP-адреси, а також використовувати засоби анонімізації даних.

						ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			10

Після впровадження програмного модулю парсингу сайту необхідно забезпечити моніторинг його роботи та вчасне виявлення та виправлення помилок. Також слід забезпечити резервне копіювання отриманої інформації, щоб уникнути її втрати у разі непередбачуваних ситуацій.

Узагальнюючи, розробка та впровадження програмного модулю парсингу сайту є складним технічним процесом, який вимагає уважної підготовки, врахування питань безпеки та правових аспектів, а також виконання вимог щодо точності та швидкості збору даних.

1.3. Технічне завдання

1. Загальні відомості

1.1. Найменування системи

1.1.1. Повне найменування системи – «AnswerParser»

1.1.2. Скорочене найменування системи – «AnswerParser»

1.2. Планові терміни початку та закінчення робіт

Початок робіт: 14.11.2022

Закінчення робіт: 17.05.2023

1.3. Порядок оформлення і пред'явлення результатів робіт

Результатом роботи є програмний модуль парсингу сайту Answer.ua, який повинен бути написаний на мові C# на платформі WPF, база даних MS SQL з Entity Framework.

Результати роботи повинні бути представлені у вигляді вихідного коду програмного модуля та документації, яка містить наступні розділи: опис завдання, опис архітектури системи, опис реалізованих функцій та їх параметрів, приклади використання програмного модуля та інструкції зі збірки та запуску.

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		11

Документація повинна бути представлена у форматі PDF або HTML.

1.4. Головний бенефіціар та потенційні користувачі системи

Головним бенефіціаром є компанія, яка замовляє розробку програмного модуля парсингу сайту Answer.ua.

Потенційними користувачами системи є споживачі, які можуть використовувати отриману інформацію для здійснення покупок на сайті Answer.ua.

2. Мета та призначення створення системи

2.1. Призначення системи

Система є програмним модулем парсингу сайту Answer.ua. Система призначена для отримання даних з сайту Answer.ua, таких як назва товару, ціна, опис, фото тощо.

2.2. Мета створення системи

Метою створення системи є автоматизація процесу отримання даних з сайту Answer.ua. Система повинна дозволити отримувати дані з сайту Answer.ua швидко та ефективно, що забезпечить зменшення часу та зусиль, необхідних для отримання цих даних вручну. Отримання даних з сайту Answer.ua дозволить компанії-замовнику розробляти більш ефективні та точні стратегії маркетингу та продажу товарів на основі аналізу цих даних.

3. Вимоги до системи

3.1. Вимоги до системи в цілому

3.1.1. Вимоги до структури та функціонування системи, перелік підсистем

3.1.1.1. Вимоги до способів і засобів інформаційного обміну між компонентами системи

Компоненти системи повинні комунікувати між собою за допомогою HTTP-запитів. Для отримання даних з сайту Answer.ua, програмний модуль

						Аркуш
						12
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 07-21.БР	

повинен використовувати бібліотеку для парсингу HTML-коду. Для зберігання отриманих даних програмний модуль повинен використовувати базу даних.

3.1.1.2. Вимоги до режимів функціонування системи

Система повинна працювати у двох режимах: ручному та автоматичному. Ручний режим дозволяє користувачу вручну запустити парсер для отримання даних з сайту Answer.ua. Автоматичний режим дозволяє програмному модулю автоматично отримувати дані з сайту Answer.ua у встановлений користувачем інтервал часу.

3.1.1.3. Вимоги до діагностування системи

Система повинна мати можливість логувати дії та помилки, що виникають під час її роботи. Система повинна мати можливість повідомляти користувача про помилки та інші проблеми, що виникають під час її роботи.

3.1.1.4. Вимоги до режимів управління системою

Користувач повинен мати можливість налаштувати параметри автоматичного режиму роботи системи, такі як інтервал часу між отриманням даних з сайту Answer.ua. Користувач повинен мати можливість зупинити або запустити роботу системи у ручному та автоматичному режимах.

3.1.2. Показники призначення

3.1.2.1. Параметри, що характеризують ступінь відповідності системи призначенням

Максимальний час на отримання та обробку даних з сайту Answer.ua не повинен перевищувати 10 секунд. Мінімальна точність отриманих даних повинна становити не менше 95%. Система повинна бути здатна парсити сайт Answer.ua на регулярній основі і забезпечувати оновлення даних не рідше ніж 1 раз на день.

3.1.2.2. Вимоги до пристосованості системи до змін

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		13

Система повинна бути здатна адаптуватися до змін у структурі та форматі даних на сайті Answer.ua. В разі змін у структурі сайту Answer.ua, система повинна бути здатна забезпечувати збір даних з нових розділів та категорій.

3.1.2.3. Вимоги до збереження працездатності системи в різних ймовірних умовах

Система повинна бути стійкою до переривань в роботі мережі Інтернет. Система повинна бути здатна відновлювати роботу в разі відмови одного з компонентів. Система повинна бути здатна зберігати логи та інформацію про помилки, що допоможе виправити проблеми, які виникнуть під час роботи системи.

3.1.3. Вимоги до надійності

3.1.3.1. Склад показників надійності до системи в цілому

Показники надійності системи включають такі складові:

- частота відмов системи;
- час відновлення після відмови;
- середній час між відмовами;
- час відновлення після відмови;
- інші вимоги, які визначаються в процесі аналізу потреб користувачів та характеру системи.

3.1.3.2. Вимоги до надійності технічних засобів і програмного забезпечення

Вимоги до надійності технічних засобів та програмного забезпечення включають:

- стійкість до помилок вхідних даних;
- стійкість до помилок в програмному забезпеченні;

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		14

- стійкість до відмов технічних засобів;
- відновлюваність після відмови;
- інші вимоги, які визначаються в процесі аналізу потреб користувачів та характеру системи.

3.1.3.3. Вимоги до методів оцінки і контролю показників надійності на різних стадіях створення системи

Вимоги до методів оцінки та контролю показників надійності на різних стадіях створення системи включають:

- тестування на рівні модуля;
- тестування на рівні інтеграції;
- тестування на рівні системи;
- проведення аналізу відмов та визначення причин їх виникнення;
- використання методів контролю коду та аналізу вразливостей програмного забезпечення;
- регулярне проведення моніторингу показників надійності та аналіз їх змін.

3.1.4. Вимоги до ергономіки та технічної естетики

Користувальницький інтерфейс має бути зрозумілим та легким у використанні для користувачів з різним рівнем технічної грамотності. Дизайн інтерфейсу має бути естетичним та забезпечувати комфортну роботу з системою. Кольорова гамма має бути сприятливою для очей та відповідати сучасним тенденціям у дизайні. Інтерфейс має мати можливість налаштування для відповідності потребам користувача.

3.1.5. Вимоги до експлуатації, технічного обслуговування, ремонту і зберігання компонентів системи

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		15

Система повинна бути легко збережена та перенесена між різними пристроями. Система повинна підтримувати самодіагностику та мати можливість автоматичної настройки та відновлення працездатності. Вимоги до зберігання даних повинні відповідати сучасним стандартам безпеки даних та конфіденційності. Вимоги до технічного обслуговування та ремонту повинні бути мінімізовані, а процедури повинні бути доступними для широкого кола спеціалістів.

3.1.6 Вимоги до захисту інформації від несанкціонованого доступу

3.1.6.1. Вимоги до інформаційної безпеки

Забезпечення конфіденційності даних, що обробляються системою. Захист системи від несанкціонованого доступу, вірусів та шкідливих програм. Забезпечення цілісності даних, що зберігаються та передаються системою. Забезпечення доступу до системи лише авторизованим користувачам. Захист від атак типу "SQL injection", "cross-site scripting" та інших подібних атак

3.1.6.2. Вимоги до антивірусного захисту

Встановлення антивірусного програмного забезпечення на всі компоненти системи. Оновлення баз даних вірусів та програмного забезпечення антивірусу з регулярністю не рідше одного разу на день. Автоматичне виявлення та блокування шкідливих програм, що можуть завдати шкоди системі.

3.1.6.3. Розмежування відповідальності ролей при доступі

Визначення рівнів доступу до системи для різних користувачів в залежності від їхніх ролей та обов'язків. Забезпечення можливості редагування та зміни рівня доступу до системи для кожного користувача окремо. Забезпечення захисту від несанкціонованого доступу до системи та

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		16

даних, що в ній зберігаються, шляхом встановлення обмежень на доступ до важливих даних та функцій системи.

3.1.7. Вимоги до захисту від впливу зовнішніх факторів

Система повинна бути захищена від несанкціонованого доступу до даних, що обробляються. Система повинна бути захищена від впливу шкідливих програм та зловмисного коду. Система повинна бути стійкою до атак з мережі Інтернет.

3.1.8. Вимоги безпеки

Система повинна мати можливість резервного копіювання даних та їх відновлення. Система повинна забезпечувати конфіденційність даних, що обробляються. Система повинна забезпечувати цілісність даних, що обробляються. Система повинна забезпечувати доступ до даних тільки авторизованим користувачам. Система повинна мати можливість реєстрації дій користувачів для забезпечення можливості виявлення випадків порушення безпеки.

3.2. Перелік підсистем системи (при наявності підсистем).

У програмному модулі парсингу сайту Answear.ua не передбачається використання підсистем, тому перелік підсистем не потрібно вказувати.

3.3. Вимоги до видів забезпечення

3.3.1. Вимоги до математичного забезпечення

Використання алгоритмів парсингу для отримання даних з сайту. Використання алгоритмів обробки даних, таких як сортування та фільтрація. Використання математичних моделей для аналізу даних та прогнозування трендів.

3.3.2. Вимоги до інформаційного забезпечення

3.3.2.1. Вимоги до складу, структури і способів організації даних в системі

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		17

Система повинна забезпечувати збір та зберігання даних, які надходять з сайту Answer.ua відповідно до встановлених правил.

Дані повинні бути структуровані та організовані у відповідні бази даних та таблиці.

Всі дані повинні бути доступні для подальшої обробки та аналізу.

Доступ до даних повинен бути обмежений відповідно до політики безпеки даних.

3.3.2.2. Вимоги до інформаційного обміну між компонентами системи

Система повинна забезпечувати швидкий та ефективний обмін даними між всіма компонентами системи.

Дані повинні передаватися за допомогою безпечного протоколу передачі даних.

Система повинна мати можливість обміну даними з іншими зовнішніми системами, якщо це необхідно.

3.3.2.3. Вимоги до інформаційної сумісності із суміжними системами

Система повинна бути сумісна з іншими системами, що використовуються в компанії.

Дані, які обмінюються між системами, повинні бути збережені та передані без помилок та відповідати встановленим стандартам обміну даними.

Система повинна бути підтримувана та оновлюватися для забезпечення сумісності з іншими системами, що використовуються в компанії.

3.3.2.4. Вимоги використання класифікаторів та уніфікованих документів

Вимоги до використання класифікаторів та уніфікованих документів повинні бути визначені з урахуванням особливостей даних, які будуть парситись з сайту Answer.ua. Наприклад, якщо потрібно парсити категорії

					<i>ДТЕУ 121 07-21.БР</i>	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		18

товарів, можуть бути використані уніфіковані документи, що містять інформацію про всі категорії товарів на сайті.

3.3.2.5. Вимоги щодо застосування систем управління базами даних

Вимоги щодо застосування систем управління базами даних (СУБД) повинні бути визначені з урахуванням обсягу даних, які будуть зберігатись. СУБД повинна бути здатна швидко і ефективно обробляти запити на отримання даних з бази.

3.3.2.6. Вимоги до структури процесу збору, обробки, передачі даних в системі представлення даних

Вимоги до структури процесу збору, обробки, передачі даних в системі представлення даних повинні бути визначені з урахуванням частоти оновлення даних на сайті Answer.ua. Процес збору даних повинен бути автоматизованим і здійснюватись з заданою періодичністю. Обробка даних повинна бути ефективною, щоб запобігти затримкам у процесі оновлення даних на сайті. Передача даних повинна бути здійснюватись з використанням безпечного протоколу зв'язку.

3.3.2.7. Вимоги до захисту даних від руйнувань при аваріях і збоях в електроживленні системи

Система повинна мати захист від несправностей в електроживленні, що можуть вплинути на цілісність даних та їх збереження.

Система повинна мати механізми збереження даних в разі відключення живлення або інших аварійних ситуацій, що можуть призвести до втрати даних.

Система повинна мати можливість автоматичного відновлення роботи та відновлення даних у разі аварійних ситуацій.

3.3.2.8. Вимоги до контролю, зберіганню, оновленню та відновленню даних

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		19

Система повинна забезпечувати контроль цілісності та достовірності даних.

Система повинна забезпечувати механізми збереження даних, що забезпечують їх доступність та конфіденційність.

Система повинна забезпечувати можливість оновлення та видалення даних відповідно до правил зберігання даних та вимог законодавства.

Система повинна забезпечувати механізми відновлення даних у разі їх втрати або пошкодження.

3.3.2.9. Вимоги до процедури надання юридичної сили документам, що продукуються технічними засобами системи

Система повинна забезпечувати можливість визначення авторства документа та його цілісності.

Система повинна забезпечувати підтвердження автентичності документа та його відповідності вимогам законодавства.

Система повинна забезпечувати можливість збереження та відтворення документа у формі, що забезпечує його відповідність вимогам законодавства.

4. Вимоги до програмного забезпечення

Мова програмування: C# на платформі WPF, база даних MS SQL з Entity Framework.

Наявність інтерфейсу користувача для введення параметрів запиту та відображення результатів.

5. Вимоги до технічного забезпечення

Операційна система: Windows або Linux.

Мінімальна кількість оперативної пам'яті: 4 ГБ.

Мінімальний обсяг вільного місця на жорсткому диску: 10 ГБ.

Наявність підключення до Інтернету для парсингу сайту.

6. Вимоги до методичного забезпечення

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		20

Наявність документації для програмного модулю.

Інструкції з використання та налагодження програмного модулю.

Приклади використання програмного модулю для різних сценаріїв запиту даних.

1.4. Висновки до розділу 1

У використанні парсингу є свої переваги та недоліки. Якщо він використовується правильно, то може допомогти в автоматизації багатьох процесів, покращенні продуктивності та збільшенні прибутку. Однак, необхідно дотримуватися правових та етичних аспектів парсингу, а також враховувати технічні перешкоди та можливі проблеми з точністю даних. Крім того, необхідно розуміти, що парсинг може бути неприйнятним для деяких сайтів, тому необхідно забезпечити належну авторизацію та дотримуватися інших правил використання, щоб уникнути правових наслідків. Для функціональних можливостей системи ми дотримуємося вимог, до яких належить: можливість збирати, зберігати, обробляти, редагувати та оновлювати дані з сайту Answer.ua у базі даних. Також повинні бути забезпечені антивірусним захистом системи згідно до вимог інформаційної безпеки.

Для захисту від впливу зовнішніх факторів ми повинні враховувати можливість негативних впливів на систему з боку користувачів або зловмисників. Вимоги до програмного забезпечення повинні передбачати використання сучасних технологій та стандартів розробки ПЗ.

Для технічного забезпечення ми повинні передбачати наявність достатніх ресурсів для ефективної роботи системи та забезпечення її надійності. Вимоги до методичного забезпечення повинні включати

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		21

документацію, необхідну для розробки, впровадження та підтримки системи. Для математичного забезпечення ми повинні передбачати використання алгоритмів та математичних методів для оптимізації роботи системи. Вимоги до складу, структури та способів організації даних в системі, інформаційного обміну між компонентами, інформаційної сумісності з суміжними системами, використання класифікаторів та уніфікованих документів повинні бути належним чином враховані.

На основі вищезгаданих вимог і висновків можна сформулювати основні вимоги до розробки програмного модулю парсингу сайту Answer.ua:

- треба забезпечити продукт захистом від атак і впливу з зовнішнього середовища, задля більшої безпеки та надійності системи.
- система повинна мати здатність до збору і обробки інформації з сайту Answer.ua, з можливістю вибору необхідної інформації, збереження та подальше використання її в роботі з системою.
- система повинна працювати чітко та за налагодженим алгоритмом, завдяки якому буде швидко та з мінімальними затримками забезпечувати доступ до даних сайту.
- для максимальної зручності споживача, система повинна мати легкий та зрозумілий інтерфейс, завдяки якому швидше та більше дізнається клієнт.
- дуже важлива та зручна функція, яка надасть змінювати масштабування, оброблювати велику кількість даних сайту, змінювати склад кількості інформації.

Отже, враховуючи всі ці вимоги та висновки, розробка програмного модулю парсингу сайту Answer.ua повинна забезпечити надійну та безпечну роботу системи, з ефективним алгоритмом збору та обробки інформації, з зручним інтерфейсом та можливістю масштабування.

					ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		22

РОЗДІЛ 2

ПРОЄКТУВАННЯ ПРОГРАМНОГО МОДУЛЮ ПАРСИНГУ САЙТУ ANSWEAR.UA

2.1. Чинники, що визначають хід і результати роботи програмного модулю парсингу сайту Answer.ua

Хід і результати роботи програмного модулю парсингу сайту Answer.ua можуть бути визначені наступними факторами (рис. 2.1).



Рис. 2.1. Фактори, що визначають хід і результати роботи програмного модулю парсингу сайту Answer.ua

Джерело: побудовано автором

					<i>ДТЕУ 121 07-21.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Зав. каф.		Криворучко О.В.		03.03.23	Програмний модуль парсингу сайту Answer.ua	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Керівник		Тищенко Д.О.		03.03.23		P2	23	48
Гарант		Рзасва С.Л.		03.03.23		Факультет інформаційних технологій 4 курс, 7 група		
Розробив		Прокопчук В.І.		03.03.23				
					<i>Проектування програмного модулю парсингу сайту Answer.ua</i>			

Одним з найважливіших факторів, які можуть вплинути на парсинг є сама структура сайту. Тому що парсеру треба знайти шлях до даних, які потрібні для аналізу, і в разі зміни цієї структури, ми не зможемо знайти дані при зборі інформації, на сторінці в списку продуктів. Для розробки парсеру необхідно заздалегідь врахувати типи сторінок, задля точної та швидкої обробки даних в програмі. Сайт може мати багато сторінок або складну структуру, а також змінювати її і тому треба це враховувати. Адже сайти прагнуть удосконалювати свої сайти і продукцію для надання послуг. Через це автор парсеру повинен також оновлювати свій продукт, щоб він функціонував й надалі. Роблячи з цього висновок, можна визначити, що треба слідкувати за сайтом та вчасно виправляти недоліки.

Обсяг даних отриманий з сайту впливає на швидкість та результати парсингу, чим більший обсяг, тим більше потрібно часу та ресурсів для отримання й обробки даних. Великою проблемою є момент коли обсяг даних настільки великий, що техніка може відмовити в процесі, або сама програма. Це відбувається внаслідок нестачі пам'яті в техніці. Таку неgodу, ми можемо виправити виставляючи максимальний ліміт даних або використовуючи паралельну обробку даних.

Для менш проблемної ситуації при використанні парсеру, ми повинні отримувати, зберігати та відправляти дані в такому форматі, аби вони підходили нам при наступному використанні процесу парсингу. Існують декілька відомих форматів, якими користуються, бо вони є зручними та простими у використанні, такі як: XML, JSON та CSV.

Кожен вибирає потрібний формат даних, згідно з тим, як саме в парсері будуть зберігатися для обробки даних, та як вони будуть використовуватися в цих процесах. Адже потрібно заздалегідь врахувати формат даних, щоб забезпечити сприятливий формат при конвертації даних.

						ДТЕУ 121 07-21.БР	Аркуш
							24
Зм.	Аркуш	№ докум	Підпис	Дата			

Якість програмного модулю є наступним фактором для результативності парсингу. Це напряму впливає на швидкодійність та отримання потрібних даних за менший проміжок часу. При недостатньо якісних умовах можуть виникати різні помилки, затримки у парсингу або неправильно отриманий формат даних.

Тест програми, дуже важлива річ, для того, щоб зменшити кількість проблем, з якими може зіткнутися користувач, цьому треба приділити час. Адже ми говоримо про якість нашого парсеру, а він повинен безперервно, максимально швидко та точно працювати. В такому разі, цим продуктом будуть більш зацікавлені, в порівнянні з неякісним, який некоректно чи неточно працює. Потрібно також слідкувати за змінами на сайті, які можуть зменшити якість продукту тим, що змінять захист від парсингу, формат даних чи структуру.

На сайті Answer.ua можуть бути встановлені правила, щодо доступу до його даних. Якщо парсер не дотримує цих правил, то сайт може заблокувати доступ до даних або взагалі заборонити парсинг.

Кожен сайт має свої правила, авторське право та правові аспекти, їх потрібно дотримуватися для законної роботи парсеру. В іншому разі, сайт має право заблокувати парсинг або доступ до своїх даних, що для нас є неприпустимим. Також для зменшення навантаження на сервер, сайт може встановити обмеження на кількість запитів від користувача за якусь одиницю часу. Для цього сайт може використовувати наведену вище систему, яка допомагає визначити хто здійснює запит, CAPTCHA. Вона пропонує виконати якусь дію, при виконанні цих дій, система пропускає користувача надалі.

Для успішного парсингу ми можемо використати допоміжні засоби, такі як спеціальні бібліотеки чи фреймворки, які нам нададуть коректну

						Аркуш
					ДТЕУ 121 07-21.БР	25
Зм.	Аркуш	№ докум	Підпис	Дата		

роботу з заголовками запитів. Ось такі дії я описав задля законної діяльності парсеру.

Частота оновлення сайту – це один з чинників, що можуть вплинути на хід та результати роботи програмного модулю парсингу Answer.ua. Сайт може, як рідко оновлюватись так і часто. При рідкому оновленню сайту, ми можемо отримувати застарілі дані, які не є актуальними в даний час. Інший момент, а саме часте оновлення може вплинути саме тим, що якщо сайт буде часто оновлюватись, то можна це не помітити та пропустити оновлення. Через це процес парсингу може некоректно передавати чи зовсім не передавати потрібні дані.

На швидкість роботи парсера, також впливає швидкість самого інтернет- з'єднання, його тип, якість, відстань до серверу. Ми можемо використовувати як пряме підключення по кабелю або Wi-Fi. Підключення кабелем надасть нам більше надійності й швидкості при парсингу, що є позитивною стороною. Але використання Wi-Fi не виглядає поганим, тому що з ним також можна парсити. Це просто як інша опція, зі своїми плюсами та мінусами. При більшій швидкості, якості, ми швидше отримаємо бажане, через швидке завантаження потрібних нам сторінок. При поганій якості та швидкості інтернету, ми маємо вірогідність, що отримаємо помилки, такі як: помилки в з'єднанні з сайтом чи відмова у відповіді через велику кількість часу очікування. Що призведе до неможливості отримання потрібних нам даних, швидко й точно. Ми можемо знизити навантаження на мережу іншими пристроями та цим надати більшу швидкість для отримання даних.

Наявність захисту від парсингу може значно вплинути на роботу парсера. Сайт може мати захист від парсингу, що може ускладнити чи унеможливити процес парсингу, через що, парсер не буде спроможний вирішити завдання, наприклад CAPTCHA. Нам в такому випадку, потрібно буде допомогти йому вирішити це, для подальшого виконання функцій

						ДТЕУ 121 07-21.БР	Аркуш
							26
Зм.	Аркуш	№ докум	Підпис	Дата			

парсингу. Блокування IP-адресів або обмеження швидкості запитів можуть спричинити затримки в роботі парсера та знизити продуктивність. Щоб побороти захист від парсингу, можна використовувати різні техніки, такі як зміна IP-адреси, використання проксі-серверів, затримки запитів тощо. Однак, варто пам'ятати, що деякі методи боротьби з захистом можуть бути незаконними або порушувати правила використання сайту, тому слід дотримуватися етичних та юридичних норм при парсингу сайтів.

Наступним, що впливає на погану роботу парсеру, це логічно, що помилки в нашій програмі. Це впливає на точність та швидкість парсингу, ми можемо зберігати чи отримувати дані в не підходячому нам форматі. Через це нам чи буде потрібно більше часу для аналізу чи отримаємо помилку в процесі. При великому обсязі даних, також можемо зловити помилку, внаслідок чого наша програма аварійно зупинить свою роботу і закриється, що зупинить наш процес. Тому завчасне оновлення, тестування допомагають нам знайти помилки в нашому програмному модулі і запобігти таких помилок.

Останнім в моєму списку факторів є методи, які використовуються для аналізу та обробки даних. Ці методи залежать від очікуваного нами результату, в виді отриманого типу даних отриманих в завданні. З цим нам допоможе налаштування наших алгоритмів для отримання, аналізу та обробки даних з сайту. Налаштував добре наші алгоритми, ми зможемо автоматично отримати потрібні нам дані в виді ціна, назва, опис і так далі. Ми повинні прагнути до швидкого та якісного парсингу, тому використовувати балансовані в цьому плані алгоритми. Адже ми не хочемо отримувати непотрібні чи недостатню кількість даних витрачаючи зайвий час.

						ДТЕУ 121 07-21.БР	Аркуш
							27
Зм.	Аркуш	№ докум	Підпис	Дата			

2.2. Інструменти розробки програмного модулю парсингу сайту Answear.ua

Для розробки програмного модулю парсингу сайту Answear.ua можна використовувати різні інструменти залежно від технологій та мов програмування, які використовуються. Ось деякі інструменти, які можуть використовуватися:

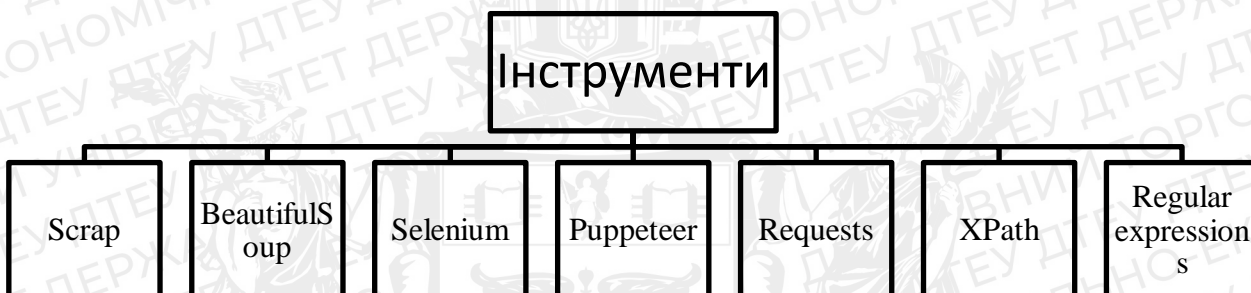


Рис. 2.2. Приклади допоміжних інструментів розробки

Джерело: побудовано автором

Для програмування парсеру на мові програмування Python ми можемо використати різні бібліотеки. Одним із них можна привести в приклад Scrapy. Він дозволяє збирати, аналізувати та обробляти дані з комфортом у виді зручного інтерфейсу з сайтів. До його можливостей входять: паралельний та асинхронний парсинг; зберігання даних таких форматів як JSON, XML та CSV; використання проксі серверів; розбір HTML-коду; обробка AJAX-запитів. Використовуючи такі можливості, ми зможемо створити програму для парсингу з сайту, отримуючи потрібні нам дані у виді – назва товару, опис, ціна та його зображення.

Розповідаючи про асинхронний парсинг можна відмітити що він дозволяє зменшити час, який витрачається на процес парсингу великого об'єму сторінок, тим що він використовує асинхронну обробку запитів для

						ДТЕУ 121 07-21.БР	Аркуш
							28
Зм.	Аркуш	№ докум	Підпис	Дата			

збору даних. Асинхронність – це можливість не чекати, поки закінчиться перше завдання, а виконувати при цьому інші. Також є інший тип парсингу – це паралельний, завдяки ньому ми маємо можливість одночасно обробляти кілька запитів.

Також допомагає використання проксі-серверів, завдяки чому, ми можемо анонімно збирати потрібні нам дані з сайтів. Розбір HTML-коду облегчить нам дійти до коду та отримати дані з допомогою іншого допоміжного інструменту розробки, BeautifulSoup. Отримання та обробка AJAX-запитів відкриє нам можливість отримати дані з сайтів, які мають ці запити для завантаження даних.

Наступною з бібліотек є BeautifulSoup, про яку ми згадували в Scrapy. Її основний козир, завдяки чому її можна використовувати, полягає в тому, що вона допомагає парсити HTML-код. Вона також присутня в мові програмування Python, як і Scrapy. Саме до її можливостей входить: розбір та маніпулювання HTML-кодом; витягувати та працювати з різними типами даних для подальшого користування, а також пошук потрібних елементів за допомогою CSS-селекторів.

Саме ця бібліотека може зробити простішим парсинг сайту саме тим, що вона зменшує кількість коду, який витягується зі зазначеного раніше HTML-коду. Як і Scrapy допоможе збирати дані з сайтів у вигляді – назва, ціна та опис товару.

Завдяки BeautifulSoup ми маємо можливість отримувати з HTML-коду такі дані, як різні зображення, посилання та текст. Також дозволяє видаляти та змінювати атрибути, змінювати HTML-код.

Наступним допоміжним інструментом, яким можна користуватися разом з наведеними раніше Scrapy та BeautifulSoup буде Selenium. Це бібліотека завдяки якій, ми отримаємо більшу точність та ефективність в програмі для парсингу сайтів. Відбувається це через те, що надає нам змогу

						ДТЕУ 121 07-21.БР	Аркуш
							29
Зм.	Аркуш	№ докум	Підпис	Дата			

виконувати різні дії на сторінках такі як, натискання кнопок, пролистування, переходи на інші сторінки та заповнення потрібних форм. Це виглядає, як користувач виконує дії на сторінці сайту і завдяки цьому ми маємо можливість отримувати з сайтів дані.

Його великим плюсом є спектр в кількості браузерів, де він може нам допомогти, в приклад я можу навести такі відомі браузери, як: Microsoft Edge, Opera, Google Chrome, Mozilla та Safari. Завдяки такій великій виборці браузерів нам можна буде виявити конкретну проблему в кожному з цих браузерів.

Схожим за застосуванням, як Selenium є інша бібліотека для мови програмування JavaScript – Puppeteer. Вона також, як і Selenium, може виконувати різні дії, такі як пролистування, переходи на інші сторінки, натискання різних кнопок на сайті та заповнення потрібних форм. Але мінусом Puppeteer є, його застосування тільки для одного браузера Google Chrome. До речі його розробниками є група робітників, які розробили сам Google Chrome. Тому в цьому плані може бути краще налаштований для парсингу з цього браузера.

З його плюсів, я можу відзначити парсинг з JavaScript-коду, отримувати URL-адрес, HTML-код. Також зчитувати текст та вміст тегів. Використовуючи цю бібліотеку, ми маємо можливість встановлювати певні налаштування браузера, а також емулювати різні девайси та записувати відео, де виконуються якісь дії на веб-сайті. Згадуючи його можливість отримувати інформацію з HTML-коду, можна сказати, що ми можемо її зберігати наприклад, в одному з цих форматів, таких як: JSON та CSV.

Допоміжний інструмент Requests допоможе нам отримувати потрібний нам HTML-код зі сторінок, авторизовуватися на різних сайтах, працювати з cookies-файлами та відправляти Post і Get запити. Ця бібліотека, як і Scrapy, Selenium та BeautifulSoup використовується у мові програмування Python.

						ДТЕУ 121 07-21.БР	Аркуш
							30
Зм.	Аркуш	№ докум	Підпис	Дата			

Вона надає нам змогу відправляти запити, такі як: Post і Get, на сервер сайту у вигляді, так званого словника. Вона, як було зазначено в можливостях, може допомогти в авторизуванні на сайтах. Для отримання зі сторінок HTML-коду, ми можемо використати метод Get, внаслідок чого він приймає URL сторінки та надає нам код у форматі текстового рядка. Також цей інструмент розробки дозволить нам налаштувати та перенаправляти заголовки запитів.

XPath – це мова запитів для вибору елементів на веб-сторінках та інших документах, що містять XML або HTML. Її можна також використовувати разом з бібліотеками наведеними раніше, такими як: Selenium, BeautifulSoup та Scrapy. Перевага цієї мови точному виборі даних, завдяки чому ми отримуємо необхідні нам дані з сайту. Вона дозволяє вибирати елементи документу за допомогою шляху до них в дереві структури документу. Шлях до елементів визначається з використанням синтаксису, що нагадує структуру файлової системи. Наприклад, шлях `"/div[@class='content']"` вибирає всі елементи div з атрибутом class, що дорівнює "content". Цей шлях може бути більш складним та включати послідовно вбудовані елементи та їх атрибути.

Regular expressions - це механізм, що використовується для пошуку та заміни тексту в рядках за допомогою відповідного шаблону. Завдяки цьому механізму ми маємо змогу отримувати інформацію з текстових даних, яку він знаходить. Як приклад можна привести таку ситуацію, ми вказуємо тег, наприклад "h", і як результат отримуємо всі елементи з цим тегом та їх атрибути. Це може бути корисним при парсингу з форматом даних, такими як: адреса електронної пошти, дата, а також номеру телефону. Необхідно зазначити, що витягування виразів, не є дуже ефективним засобом витягувати дані з HTML-коду. Це може створити деякі проблеми для новачків в цій

						ДТЕУ 121 07-21.БР	Аркуш
							31
Зм.	Аркуш	№ докум	Підпис	Дата			

справі, адже це є складним процесом для розуміння та пристосуванню цього інструменту.

Вибір інструментів для розробки програмного модулю парсингу сайту Answer.ua залежить від конкретних потреб проекту, технічних обмежень та власних знань розробника.

Що стосується «AnswerParser», то даний програмний модуль був розроблений на платформі WPF, база даних MS SQL з Entity Framework. Тому розглянемо також даний інструмент.

WPF є технологією, яка використовується для створення клієнтських додатків для Windows, які мають графічний інтерфейс користувача. Вона пропонує багато можливостей для створення привабливого інтерфейсу з використанням різних ефектів та анімацій. WPF підтримує такі мови програмування, як C# і VB.NET.

MS SQL є однією з найпоширеніших систем управління базами даних. Вона забезпечує зберігання, керування та доступ до даних з використанням мови SQL.

Entity Framework є технологією ORM, яка дозволяє взаємодіяти з базою даних, використовуючи об'єктно-орієнтовані класи. За допомогою EF можна звертатися до бази даних, виконувати запити та модифікувати дані безпосередньо з коду. EF підтримує багато типів баз даних, включаючи MS SQL.

Для розробки програмного модулю парсингу сайту Answer.ua, можна використати WPF, MS SQL та Entity Framework разом.

Спочатку, необхідно створити базу даних у MS SQL, в якій будуть зберігатися дані, що отримуються під час парсингу.

Далі, можна створити проект WPF, де буде відображатися інформація, що була зібрана з сайту Answer.ua. В проекті WPF можна використовувати EF для підключення до бази даних та виконання запитів.

						ДТЕУ 121 07-21.БР	Аркуш
							32
Зм.	Аркуш	№ докум	Підпис	Дата			

Після цього, можна написати код, який буде парсити сайт Answear.ua та збирати необхідну інформацію. Отримані дані можна зберігати в базі даних за допомогою EF.

Для покращення швидкодії та ефективності роботи, можна використовувати пакети NuGet, наприклад, HtmlAgilityPack для парсингу HTML, або Microsoft.Extensions.DependencyInjection для реалізації залежностей.

Також, можна використовувати різноманітні патерни проектування, такі як Singleton, Facade, або Factory, щоб полегшити розробку та покращити структуру коду.

Після того, як дані збережено в базі даних, можна відображати їх у програмі, використовуючи WPF. Для цього можна створити відповідні візуальні елементи, такі як таблиці, списки або графіки. Дані можна відображати в режимі реального часу, оновлюючи їх після кожного нового парсингу сайту.

Крім того, можна додати функції експорту та імпорту даних з програми до файлів, щоб забезпечити зручність користувачів у використанні даних.

Загалом, використання WPF, MS SQL з Entity Framework дозволяє створити ефективний та функціональний програмний модуль парсингу сайту Answear.ua, який зможе забезпечити користувачам зручний та надійний доступ до необхідної інформації.

2.3. Висновки до розділу 2

Інструменти розробки програмного модулю парсингу сайту Answear.ua можуть бути використані для отримання даних зі складних веб-сторінок.

BeautifulSoup, Scrapy, Requests, XPath та регулярні вирази є популярними інструментами, які можуть бути використані для парсингу веб-

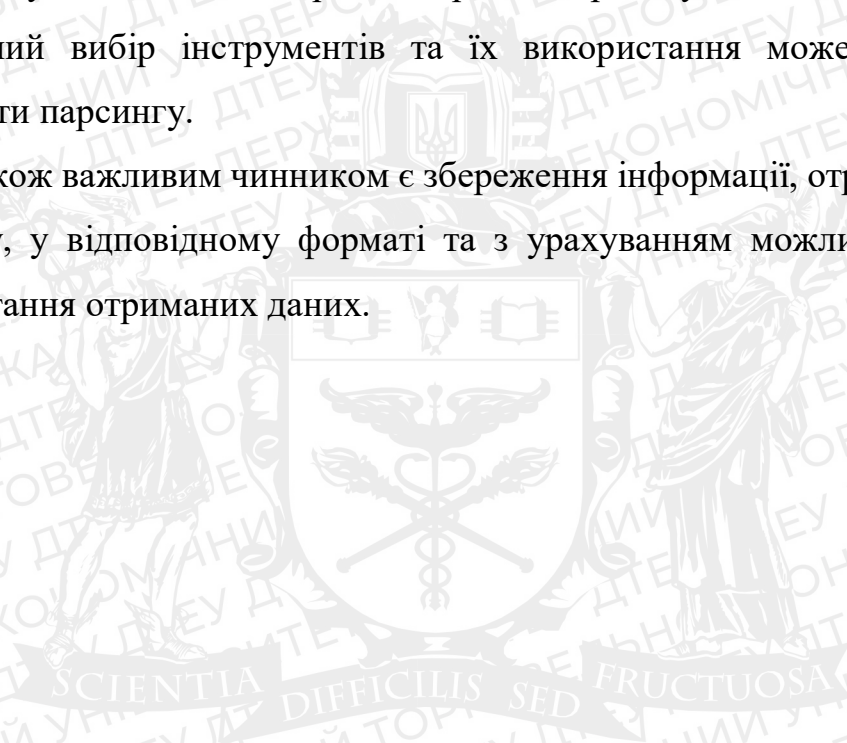
						ДТЕУ 121 07-21.БР	Аркуш
							33
Зм.	Аркуш	№ докум	Підпис	Дата			

сторінок. Вони дозволяють отримати доступ до HTML-коду сторінок та витягнути з них різні дані.

Крім цього, Selenium та Puppeteer можуть бути використані для автоматизації дій на веб-сторінках та отримання даних з них.

Хід і результати роботи програмного модулю парсингу сайту Answer.ua залежать від декількох факторів. Серверна сторона веб-сайту може блокувати запити, що робить процес парсингу складнішим. Крім того, правильний вибір інструментів та їх використання може вплинути на результати парсингу.

Також важливим чинником є збереження інформації, отриманої під час парсингу, у відповідному форматі та з урахуванням можливих обмежень використання отриманих даних.



						ДТЕУ 121 07-21.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			34

РОЗДІЛ 3

РОЗРОБКА ТА ВПРОВАДЖЕННЯ ПРОГРАМНОГО МОДУЛЮ ПАРСИНГУ САЙТУ ANSWEAR.UA

3.1. Функціональні характеристики розробленого програмного модулю парсингу сайту Answer.ua

Програмний модуль парсингу Answer.ua має наступні функціональні характеристики:

- може завантажувати сторінки сайту Answer.ua для подальшого аналізу.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Runtime.CompilerServices;
using System.Text;
using System.Threading.Tasks;

namespace ParserClient.ViewModels
{
    public class ViewModelBase : INotifyPropertyChanged
    {
        public event PropertyChangedEventHandler PropertyChanged;

        protected virtual bool SetProperty<T>(ref T storage, T value, [CallerMemberName] string propertyName = null)
        {
            if (Equals(storage, value)) return false;
            storage = value;
            OnPropertyChanged(propertyName);
            return true;
        }
        protected virtual void OnPropertyChanged([CallerMemberName] string propertyName = null)
        {
            var eventHandler = PropertyChanged;
            if (eventHandler != null)
                eventHandler(this, new PropertyChangedEventArgs(propertyName));
        }
    }
}
```

Рис. 3.1. Завантаження веб-сторінок

Джерело: побудовано автором

					<i>ДТЕУ 121 07-21.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Зав. каф.		Криворучко О.В.		14.04.23	Програмний модуль парсингу сайту Answer.ua	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Керівник		Тищенко Д.О.		14.04.23		<i>РЗ</i>	<i>35</i>	<i>48</i>
Гарант		Рзаєва С.Л.		14.04.23		Факультет інформаційних технологій 4 курс, 7 група		
Розробив		Прокопчук В.І.		14.04.23				
					Розробка та впровадження програмного модулю парсингу сайту Answer.ua			

- може аналізувати HTML-код сторінок для вилучення необхідної інформації. Наприклад, програмний модуль може аналізувати теги HTML, класи CSS, атрибути і текстовий контент.

```

using AnswerParser;
using AnswerParser.Models;
using ParserClient.Data;
using ParserClient.Models;
using System;
using System.Collections.Generic;
using System.Collections.ObjectModel;
using System.Diagnostics;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;

namespace ParserClient.ViewModels
{
    public class ParseByParamsVM : ViewModelBase, IPageViewModel
    {
        public List<Category> Categories { get; set; }

        public List<Gender> Genders { get; set; }
        public string Name => "StoreSearchOutline";

        public ParseByParamsVM()
        {
            Categories = new List<Category>
            {
                new Category() { LogicalName = "odyag", Name = "Одяг" },
                new Category() { LogicalName = "vzuttya", Name = "Взуття" },
                new Category() { LogicalName = "aksesuary", Name = "Акcesуари" },
                new Category() { LogicalName = "odyag", Name = "Одяг" }
            };
            Genders = new List<Gender>
            {
                new Gender() { LogicalName = "Women", Name = "Жіночий" },
                new Gender() { LogicalName = "Men", Name = "Чоловічий" }
            };
            PageCount = 1;
            Products = new ObservableCollection<Product>();
        }
    }
}

```

Рис. 3.2. Аналіз HTML-коду сторінок

Джерело: побудовано автором

- може вилучати різноманітні дані зі сторінок Answer.ua, такі як ціни, назви товарів, описи товарів, зображення і т.д.
- може зберігати вилучені дані в різних форматах, наприклад, в базі даних або в CSV-файлах.
- може проводити обробку вилучених даних, наприклад, проводити аналіз даних, відбирати дані за певними критеріями, і т.д.

						Аркуш
						36
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 07-21.БР	

- може виконувати автоматичні завдання, такі як поновлення цін на товари або відстеження змін в продуктовому асортименті Answer.ua.

```
using AnswerParser.Models;
using System;
using System.Collections.Generic;
using System.Collections.ObjectModel;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Xml.Serialization;

namespace ParserClient.Models
{
    public class FileSystem
    {
        public static void SaveToFile(List<Product> Products)
        {
            var dialog = new Microsoft.Win32.SaveFileDialog();
            dialog.Filter = "XML files (*.xml)|*.xml|All files (*.*)|*.*";
            if (dialog.ShowDialog() == true)
            {
                string filePath = dialog.FileName;
                // збереження даних в файл по шляху filePath
                var serializer = new XmlSerializer(typeof(List<Product>));
                using (var writer = new StreamWriter(filePath))
                {
                    serializer.Serialize(writer, Products);
                }
            }
        }
    }
}
```

Рис. 3.3. Автоматизація завдань

Джерело: побудовано автором

- може бути налаштований для обмеження швидкості запитів до сайту Answer.ua, щоб уникнути перевантаження серверів.
- може використовувати проксі-сервери для захисту від блокування IP-адрес.
- може використовувати API Answer.ua для отримання даних з сайту.
- може аналізувати динамічний контент, що генерується JavaScript, для отримання додаткової інформації.
- може використовувати многопоточність для прискорення процесу парсингу.

						Аркуш
						37
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 07-21.БР	

- може включати функції обробки помилок, щоб забезпечити стабільну роботу програмного забезпечення.

Загалом, результати роботи програмного модулю парсингу сайту Answer.ua залежать від багатьох факторів, тому важливо враховувати всі ці фактори при розробці та використанні парсера. Головною метою мого продукту є автоматизація в зборі даних з сайту Answer.ua для зручного використання парсеру.

3.2. Особливості впровадження програмного модулю парсингу сайту Answer.ua

Впровадження програмного модулю парсингу сайту Answer.ua може мати кілька особливостей, які можуть впливати на процес розробки і використання модулю. Нижче наведено деякі з можливих особливостей впровадження програмного модулю парсингу Answer.ua (рис. 3.4).

						Аркуш
					ДТЕУ 121 07-21.БР	38
Зм.	Аркуш	№ докум	Підпис	Дата		



Рис. 3.4. Особливості впровадження програмного модулю парсингу сайту Answer.ua

Джерело: побудовано автором

Першою із особливостей впровадження є блокування IP-адрес. Цей процес відбувається внаслідок великої кількості запитів з одного IP-адресу зі сторони користувача. Оскільки це може вплинути на працездатність сайту внаслідок перенавантаження серверу або зробити сайт недоступним на якийсь момент часу. Щоб цього уникнути сайт використовує таку можливість.

Ми можемо використовувати різні IP-адреса за допомогою проксі-серверів. Проксі-сервер є посередником в цій справі, це спричинить меншу кількість запитів на сервер і таким чином ми обійдемо блокування IP-адресу.

Також ми можемо використати допоміжні інструменти, які нададуть нам варіації обходу, такі як: зменшення швидкості запитів; використання cookies-файлів; зміна заголовку запиту; відправлення запитів різними IP-адресами, які змінюються через якийсь інтервал. Дані інструменти та використання проксі-серверів нам допоможуть обійти блокування нашої адреси та подальше використання парсеру на сайті Answer.ua.

Для отримання даних з нашого сайту, ми використаємо API. Він нам надасть змогу організувати взаємодію між програмою та сайтом Answer.ua. Для продуктивного використання, що надасть нам більше надійності та швидкості отримання даних, ми повинні його належно використовувати. Тому що він має обмеження на кількість запитів за одну спробу.

Ми можемо використати даний інтерфейс для отримання даних з списку товарів, замість парсингу сторінки. Адже це дасть нам змогу збільшити швидкість на отримання даних сайту та стабільність отримання потрібних нам даних. Використовуючи цей інструмент, він нам дозволить отримувати потрібні дані з сайту завдяки доступу до спеціально розроблених інтерфейсів.

Наступним процесом після отримання потрібних даних з сайту Answer.ua, у нас настає процес обробки даних. Цей процес має такі кроки, а саме: збереження даних з сайту до бази даних; аналіз даних; створення звітів та згрупування отриманих даних за заданим шаблоном.

Нашою метою є зберігання даних товарів, для досягнення цієї мети потрібна база даних де вона буде зберігатись для використання у майбутньому. Нам потрібно забезпечити інтеграцію з іншими додатками, серед таких є СУБД чи засіб аналізу даних, що надасть змогу нам зберігати дані до неї. Для належної роботи програми ми можемо виконати такі функції як: конвертація даних до потрібного формату, валідація та процес очищення даних перед збереженням.

Обмеження прав доступу є важливим аспектом забезпечення безпеки та захисту конфіденційної інформації на сайтах, в тому числі і на Answer.ua. Це означає, що сайт надає спеціальні можливості для певних користувачів на певних сторінках. Таке сайт може використовувати для обмеження доступу небажаних лиць, адже тільки спеціальні люди повинні мати доступ до конфіденційної інформації, контролю над функціональністю сайту внаслідок

						ДТЕУ 121 07-21.БР	Аркуш
							40
Зм.	Аркуш	№ докум	Підпис	Дата			

зовнішніх факторів впливу на сервера сайту. До конфіденційної інформації сайту належать наприклад: логін та пароль користувача, його персональні дані.

Сайт повинен мати різні рівні доступу користувачів до сторінок сайту. Цього досягти допоможуть різні системи для керування доступом та системи авторизації з ідентифікацією користувача. На розвинених сайтах, задля безпеки користувачів, може використовуватися так звана, двофакторна автентифікація, яка просить підтвердити захід у профіль користувача, наприклад введенням коду з програми, яка його рандомно генерує, що надає нам більшу безпеку нашого профілю.

Зміна HTML-структури може стати проблемою у процесі парсингу даних з сайту. Адже власники сайту прагнуть удосконалювати і це призводить до змін, що може надати нам таку проблему. Внаслідок цього парсер має можливість не знайти необхідні йому елементи на сайті, що призведе до незадовільного результату у вигляді неправильних чи пошкоджених даних. Щоб цього уникнути у нас буде необхідність переробити наш парсер, а саме змінивши алгоритм пошуку потрібного на сайті та додавши додаткові логічні перевірки.

Також можуть бути введені додаткові захисні механізми на сайті Answear.ua для запобігання парсингу даних, наприклад, використання CAPTCHA або захисту від ботів. У такому випадку може бути необхідно знайти спосіб обійти ці захисні механізми, щоб збирати необхідні дані з сайту Answear.ua.

Коли вимоги до швидкості є критичними, програмний модуль парсингу Answear.ua може бути оптимізований, щоб прискорити процес отримання даних. Для цього можуть бути використані наступні підходи: кешування даних, мінімізація кількості запитів до сервера, використання інструментів

						Аркуш
					ДТЕУ 121 07-21.БР	41
Зм.	Аркуш	№ докум	Підпис	Дата		

для швидкої обробки даних, асинхронних запитів та оптимальних налаштувань.

При рідкісній зміні даних сайту, ми можемо зберігати копію отриманих даних внаслідок парсингу для використання їх у майбутніх запитах, задля економії часу. Для більшої швидкості, якщо існує можливість отримати потрібні нам дані за один раз, треба це використати, щоб зменшити час, який витрачається на збір даних.

Також для більш швидкого процесу потрібно використовувати найбільш швидкі алгоритми та технології для парсингу, але щоб при цьому не постраждала його точність. Використовуючи такі технології, як асинхронні запити, паралельна обробка даних чи його векторизація. Асинхронні запити використовуються для пришвидшення, їх функція в одночасній відправці та очікуванні запитів.

Одним із найважливіших при використанні парсеру, є дотримання користувача до правил сайту. Для цього потрібно не навантажувати сервера, відправляючи багато запитів за малу кількість часу. Потрібно дотримуватися до політики використання даних сайту, і обмеження сайтом для коректного використання для захисту його від зовнішніх атак, наприклад DDoS-атак.

Для дотримання правил сайту можна спробувати використати спеціальні заголовки запитів, які допоможуть нам в ідентифікуванні, наприклад, User-Agent, що ідентифікує програмний модуль парсингу, щоб сайт Answer.ua міг відрізнити його від інших запитів та застосовувати до нього належні обмеження. Якщо порушити політику то це може призвести до несення відповідальності, або відшкодування збитків сайту за такий інцидент чи до блокування та недоступності користувачами до сайту. Тому дуже важливо дотримуватися правил сайту та використовувати програмний модуль парсингу відповідально та з урахуванням правил та обмежень, що на нього накладаються.

						Аркуш
					ДТЕУ 121 07-21.БР	42
Зм.	Аркуш	№ докум	Підпис	Дата		

Якщо програмний модуль парсингу Answer.ua використовується в комплексі з іншими інструментами, то можуть виникати деякі вимоги до сумісності з ними. Наприклад, якщо дані збираються з метою збереження до бази даних, то важливо, щоб програмний модуль парсингу міг коректно взаємодіяти з базою даних та записувати дані у відповідному форматі.

3.3. Висновки до розділу 3

Для забезпечення сумісності програмного модулю парсингу з іншими інструментами може знадобитися додатковий розвиток та налаштування програми. А саме встановлення додаткових інструментів для інтеграції з базою даних.

Крім того, важливо враховувати можливі конфлікти між програмним модулем парсингу Answer.ua та іншими інструментами, які можуть використовуватися на тому ж сервері або комп'ютері. Наприклад, можуть виникати конфлікти з пам'яттю або з портами, через які взаємодіють інструменти. Для забезпечення сумісності може бути необхідно виконати додаткову настройку або розширення ресурсів сервера.

Мій програмний модуль парсингу може бути корисним для бізнесу, який займається продажем товарів у соціальних мережах або на інших онлайн платформах. Завдяки ньому маємо можливість точно та швидко отримати дані про товар взятий з сайту Answer.ua. Це можна використати наприклад для аналізу цін на товари, якоїсь важливої інформації.

Для законного використання цього модулю ми повинні дотримуватися до законодавства та політики сайту. Для актуальності його оновлювати внаслідок змін, тим самим забезпечивши його ефективність та безпеку.

						Аркуш
					ДТЕУ 121 07-21.БР	43
Зм.	Аркуш	№ докум	Підпис	Дата		

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

У використанні парсингу є свої переваги та недоліки. Якщо він використовується правильно, то може допомогти в автоматизації багатьох процесів, покращенні продуктивності та збільшенні прибутку. Однак, необхідно дотримуватися правових та етичних аспектів парсингу, а також враховувати технічні перешкоди та можливі проблеми з точністю даних. Крім того, необхідно розуміти, що парсинг може бути неприйнятним для деяких сайтів, тому необхідно забезпечити належну авторизацію та дотримуватися інших правил використання, щоб уникнути правових наслідків.

Вимоги до функціональних можливостей системи повинні включати в себе збір, обробку та зберігання даних, що отримуються з веб-сторінок Answear.ua. Також необхідно передбачити можливість редагування та оновлення даних у базі даних системи. Вимоги до інформаційної безпеки та антивірусного захисту системи повинні бути належним чином забезпечені.

Вимоги до захисту від впливу зовнішніх факторів повинні враховувати можливість негативних впливів на систему з боку користувачів або зловмисників. Вимоги до програмного забезпечення повинні передбачати використання сучасних технологій та стандартів розробки ПЗ.

Вимоги до технічного забезпечення повинні передбачати наявність достатніх ресурсів для ефективної роботи системи та забезпечення її надійності. Вимоги до методичного забезпечення повинні включати документацію, необхідну для розробки, впровадження та підтримки системи. Вимоги до математичного забезпечення повинні передбачати використання алгоритмів та математичних методів для оптимізації роботи системи. Вимоги до складу, структури та способів організації даних в системі, інформаційного

Зм.	Аркуш	№ докум.	Підпис	Дата	<i>ДТЕУ 121 07-21.БР</i>			
Зав. каф.		Криворучко О.В.		28.04.23				Програмний модуль парсингу сайту Answear.ua
Керівник		Тищенко Д.О.		28.04.23	ВП	44	48	
Гарант		Рзаєва С.Л.		28.04.23	Факультет інформаційних технологій			
Розробив		Прокопчук В.І.		28.04.23	4 курс, 7 група			
					Висновки та пропозиції			

обміну між компонентами, інформаційної сумісності з суміжними системами, використання класифікаторів та уніфікованих документів повинні бути належним чином враховані.

На основі вищезгаданих вимог і висновків можна сформулювати основні вимоги до розробки програмного модулю парсингу сайту Answer.ua:

- система повинна мати високу надійність, безпеку, захист від впливу зовнішніх факторів та інформаційної безпеки, а також захист від атак з зовнішнього середовища.
- система повинна мати здатність до збору і обробки інформації з сайту Answer.ua, з можливістю вибору необхідної інформації, збереження та подальше використання її в роботі з системою.
- система повинна працювати швидко, з мінімальними часовими затримками, забезпечувати швидкий доступ до даних та оптимізований алгоритм обробки.
- система повинна бути зрозумілою та зручною в користуванні, з легким інтерфейсом та інтуїтивно зрозумілими функціями.
- система повинна мати можливість масштабування, з розширенням до інших сайтів, можливість змінювати склад та кількість інформації, оброблювати великий обсяг даних.

Отже, враховуючи всі ці вимоги та висновки, розробка програмного модулю парсингу сайту Answer.ua повинна забезпечити надійну та безпечну роботу системи, з ефективним алгоритмом збору та обробки інформації, з зручним інтерфейсом та можливістю масштабування.

Для забезпечення сумісності програмного модулю парсингу з іншими інструментами може знадобитися додатковий розвиток та налаштування програми. Наприклад, може бути необхідним встановлення додаткових модулів для забезпечення інтеграції з базою даних або іншими інструментами.

						ДТЕУ 121 07-21.БР	Аркуш
							45
Зм.	Аркуш	№ докум	Підпис	Дата			

Крім того, важливо враховувати можливі конфлікти між програмним модулем парсингу Answer.ua та іншими інструментами, які можуть використовуватися на тому ж сервері або комп'ютері. Наприклад, можуть виникати конфлікти з пам'яттю або з портами, через які взаємодіють інструменти. Для забезпечення сумісності може бути необхідно виконати додаткову настройку або розширення ресурсів сервера.

Загалом, впровадження програмного модулю парсингу сайту Answer.ua може бути корисним для бізнесу, який займається продажем товарів у соціальних мережах або на інших онлайн платформах. Він дозволяє швидко та ефективно отримувати актуальну інформацію про товари та їх характеристики на сайті Answer.ua. Програмний модуль може бути використаний для виконання різних завдань, таких як моніторинг цін на товари, отримання інформації про товари для аналізу ринку, створення власних каталогів товарів та багато іншого.

Проте, при впровадженні програмного модулю парсингу Answer.ua необхідно дотримуватися різних вимог та виконувати різні завдання для забезпечення його ефективності та безпеки. Крім того, важливо бути уважним до законодавчих обмежень, які стосуються використання даних та захисту персональних даних.

						Аркуш
						46
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 07-21.БР	

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Драч В.Є., Родіонов А.В., Чухраєва А.І. Вибір системи управління базами даних для інформаційної системи промислового підприємства // Електромагнітні хвилі і електронні системи. – 2018.
2. Тищенко Д.В. Автоматизація обліку навчального навантаження на кафедрі. / Тищенко Д.В., Ващенко С.М. // Інформатика, математика, автоматика : матеріали та програма науково-технічної конференції, м. Суми, 2019 р. – Суми : СумДУ, 2019.
3. Шендрик В.В. Система збирання, розміщення та аналізу даних [Текст] / В.В. Шендрик, С.М. Ващенко // Вісник Національного університету "Львівська політехніка". - 2011. - № 715. - С. 1-11.
4. Adobe. What is PDF? [Електронний ресурс] / Adobe – Режим доступу до ресурсу: <https://acrobat.adobe.com/us/en/acrobat/about-adobe-pdf.html>.(дата звернення: 20.04.2023)
5. Aho A. V. The theory of parsing, translation, and compiling / A. V. Aho, J. D. Ullman. – USA: Prentice-Hall, 1972. – 121 с.
6. AngularJS [Електронний ресурс] – Режим доступу до ресурсу: <https://angularjs.org/>.(дата звернення: 20.04.2023)
7. Aristarán M. Tabula [Електронний ресурс] / M. Aristarán, M. Tigas, J. Merrill – Режим доступу до ресурсу: <https://tabula.technology/>.(дата звернення: 20.04.2023)
8. Beautiful Soup 4 Python [Електронний ресурс] – Режим доступу до ресурсу: <http://www.pythonforbeginners.com/beautifulsoup/beautifulsoup-4-python>. (дата звернення: 26.04.2023)

Зм.	Аркуш	№ докум.	Підпис	Дата	<i>ДТЕУ 121 07-21.БР</i>			
Зав. каф.		Криворучко О.В.		23.12.22	Програмний модуль парсингу сайту Answer.ua	Стадія	Аркуш	Аркушів
Керівник		Тищенко Д.О.		23.12.22		СВД	47	48
Гарант		Рзаєва С.Л.		23.12.22		Факультет інформаційних технологій 4 курс, 7 група		
Розробив		Прокопчук В.І.		23.12.22				
<i>Список використаних джерел</i>								

9. Downey A. B. Think Java How to Think Like a Computer Scientist / Allen Downey., 2012
10. Eric Elliott. Programming JavaScript Applications: Robust Web Architecture with Node, HTML5, and Modern JS Libraries. – O'Reilly Media, 2014. – 254 с.
11. Eric Freeman, Elisabeth Robson. Head First HTML and CSS. – O'Reilly Media, 2005. – 694 с.
12. Grune D. Parsing Techniques - A Practical Guide / D. Grune, C. Jacobs. – Chichester: Originally published by Ellis Horwood, 1990. – 320 с.
13. Jennifer Niederst. Web Design in a Nutshell: A Desktop Quick Reference. – O'Reilly Media, 2006. – 826 с.
14. Lxml - XML and HTML with Python [Електронний ресурс] – Режим доступу до ресурсу: <http://lxml.de/>. (дата звернення: 29.04.2023)
15. Miles R. C# Programming Yellow Book / Rob Miles., 2015. С. 14.
16. Mitchell R. Web Scraping with Python / Ryan Mitchell. – Boston: O'Reilly Media, 2015. – 256 с.
17. Parse [Електронний ресурс]. – 2010. – Режим доступу до ресурсу: <https://www.dictionary.com/browse/parse>. (дата звернення: 30.04.2023)
18. Popular Technology. Режим доступу: <http://www.populartechology.net/2010/08/google-scholar-illiteracy-in-pnas.html>. (дата звернення: 05.05.2023)
19. Rouse M. Portable Document Format (PDF) [Електронний ресурс] / Margaret Rouse. – 2010. – Режим доступу до ресурсу: <https://whatis.techtarget.com/definition/Portable-Document-Format-PDF>. (дата звернення: 06.05.2023)

					<i>ДТЕУ 121 07-21.БР</i>			
Зм.	Аркуш	№ докум.	Підпис	Дата				
Зав. каф.		Криворучко О.В.		23.12.22	Програмний модуль парсингу сайту Answer.ua	Стадія	Аркуш	Аркушів
Керівник		Тищенко Д.О.		23.12.22		СВД	48	48
Гарант		Рзаєва С.Л.		23.12.22		Факультет інформаційних технологій 4 курс, 7 група		
Розробив		Прокопчук В.І.		23.12.22				
					<i>Список використаних джерел</i>			

ДОДАТКИ

ДОДАТОК А

Код ProductsViewModel

```
using AnswearParser.Models;
using Microsoft.EntityFrameworkCore;
using ParserClient.Data;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;

namespace ParserClient.ViewModels
{
    public class ProductsViewModel : ViewModelBase, IPageViewModel
    {
        public string Name => "ListBoxOutline";

        private List<Product> _products;

        public List<Product> Products {
            get => _products;
            set
            {
                _products = value;
                OnPropertyChanged();
            }
        }

        public ProductsViewModel()
        {
            Products = new List<Product>();
            using (var db = new DbContext())
            {
                Products = db.Products.ToList();
            }
        }
    }
}
```

Код класу Product

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;
```

```
namespace AnswerParser.Models
```

```
{  
    public class Product  
    {  
        public int Id { get; set; }  
        public string Name { get; set; }  
        public string Url { get; set; }  
        public string Price { get; set; }  
        public string ImageUrl { get; set; }  
        public string Description { get; set; }  
    }  
}
```



Код Parser

```

using AnswerParser.Models;
using HtmlAgilityPack;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using static System.Runtime.InteropServices.JavaScript.JSType;

namespace AnswerParser
{
    public class Parser
    {
        public string url = "https://answer.ua/k/vin/odyag";
        public Parser() { }

        public List<Product> ParseFromUrl(string Url, int pageCount)
        {
            try
            {
                var urls = parseItemUrls(Url, pageCount);
                return (ParseInfoFromList(urls));
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
                return new List<Product> { };
            }
        }

        public List<Product> ParseWithParams(string Gender, int pageCount = 1)
        {
            if (!string.IsNullOrEmpty(Gender) & new string[] { "Women", "Men" }.Contains(Gender))
            {
                switch (Gender)
                {
                    case "Women":
                        var urls = parseItemUrls("https://answer.ua/k/vona/odyag", pageCount);
                        return(ParseInfoFromList(urls));
                    case "Men":
                        var urls = parseItemUrls("https://answer.ua/k/vin/odyag", pageCount);
                        return (ParseInfoFromList(urls));
                    default:
                        return new List<Product> { };
                }
            }
            else
            {
                throw new Exception("Bad Input Params");
            }
        }

        public async Task<List<Product>> ParseWithParamsAsync(string Gender, IProgress<int> progressIndicator, string Category, int
        pageCount = 1)
        {
            if (!string.IsNullOrEmpty(Gender) & new string[] { "Women", "Men" }.Contains(Gender))
            {
                switch (Gender)
                {
                    case "Women":
                        var urls = await
                        ParseItemUrlsAsync($"https://answer.ua/k/vona/{Category}", pageCount);
                        return (await ParseInfoFromListAsync(urls, progressIndicator));
                    case "Men":
                        var urls = await ParseItemUrlsAsync($"https://answer.ua/k/vin/{Category}",
                        pageCount);
                        return (await ParseInfoFromListAsync(urls, progressIndicator));
                }
            }
        }
    }
}

```

default:

Продовження додатку В

```
return new List<Product> { };
}
}
else
{
    throw new Exception("Bad Input Params");
}
}
private List<string> parseItemUrls(string url, int pageCount = 1)
{
    List<string> urls = new List<string>();
    HtmlWeb web = new HtmlWeb();
    HtmlDocument doc = web.Load(url);

    foreach (HtmlNode productNode in
        doc.DocumentNode.SelectNodes("//div[@class='ProductItem__productCardImageWrapper__8ls-g']/a"))
    {
        Console.WriteLine(productNode);
        urls.Add(productNode.Attributes["href"].Value);
    }
    return urls;
}

private List<Product> ParseInfoFromList(List<string> itemUrls)
{
    List<Product> products = new List<Product>();
    HtmlWeb web = new HtmlWeb();
    foreach (string url in itemUrls)
    {
        try {
            HtmlDocument doc = web.Load("https://answear.ua" + url);
            HtmlNode priceNode = doc.DocumentNode.SelectSingleNode("//div[@class='Price__price__CbfdW
                ProductCard__priceRegular__sFepg']");

            Product product = new Product();
            if (priceNode != null) {
                product.Price = priceNode.InnerText;
            }
            else
            {
                priceNode = doc.DocumentNode.SelectSingleNode("//div[@class='Price__price__CbfdW
                Price__priceWithSale__FyecW ProductCard__priceRegularWithSale__jhCPs']");
                product.Price = priceNode.InnerText;
            }

            HtmlNode descriptionNode =
                doc.DocumentNode.SelectSingleNode("//div[@class='Accordion__accordionDescription__CckLI']");
            var description = descriptionNode.ChildNodes.Select(x => x.InnerText);
            product.Description = string.Join(" ", description);
            HtmlNode nameNode =
                doc.DocumentNode.SelectSingleNode("//div[@class='ProductCard__productNameAndLogo__oKv+O']");
            product.Name = nameNode.InnerText;
            product.Url = url;
            HtmlNode imageNode = doc.DocumentNode.SelectSingleNode("//picture[@class='Image__cardImage__xvgs1']");
            product.ImageUrl = imageNode.ChildNodes[1].Attributes["src"].Value;

            products.Add(product);
        }
        catch (Exception ex) {
            Console.WriteLine(ex.Message);
        }
    }
    return products;
}

//async

public async Task<List<Product>> ParseFromUrlAsync(string url, int pageCount, IProgress<int> progress = null)
{
    try
    {
        var urls = await ParseItemUrlsAsync(url, pageCount);
        return await ParseInfoFromListAsync(urls, progress);
    }
    catch (Exception ex)
    {
        Console.WriteLine(ex.Message);
        return new List<Product>();
    }
}

private async Task<List<string>> ParseItemUrlsAsync(string url, int pageCount = 1)
{
    List<string> urls = new List<string>();
    HtmlWeb web = new HtmlWeb();
    for (int i = 1; i <= pageCount; i++)
```


Продовження додатку В

```
{
    string pageUrl = $"{url}?page={i}";

    HtmlDocument doc = await web.LoadFromWebAsync(pageUrl);
    foreach (HtmlNode productNode in
        doc.DocumentNode.SelectNodes("//div[@class=ProductItem__productCardImageWrapper__8Is-g]/a"))
    {
        urls.Add(productNode.Attributes["href"].Value);
    }
}
return urls;
}

private async Task<List<Product>> ParseInfoFromListAsync(List<string> itemUrls, IProgress<int> progress = null)
{
    List<Product> products = new List<Product>();
    HtmlWeb web = new HtmlWeb();
    int count = itemUrls.Count;
    for (int i = 0; i < count; i++)
    {
        try
        {
            HtmlDocument doc = await web.LoadFromWebAsync($"https://answear.ua{itemUrls[i]}");
            HtmlNode priceNode =
                doc.DocumentNode.SelectSingleNode("//div[@class=ProductCard__priceRegular__sFepg]");
            Product product = new Product();
            if (priceNode != null)
            {
                product.Price = priceNode.InnerText;
            }
            else
            {
                priceNode =
                    doc.DocumentNode.SelectSingleNode("//div[@class=ProductCard__priceRegularWithSale__jhCPs]");
                product.Price = priceNode.InnerText;
            }
            HtmlNode descriptionNode =
                doc.DocumentNode.SelectSingleNode("//div[@class=Accordion__accordionDescription__CckLI]");
            var description = descriptionNode.ChildNodes.Select(x => x.InnerText);
            product.Description = string.Join(";", description);
            HtmlNode nameNode =
                doc.DocumentNode.SelectSingleNode("//div[@class=ProductCard__productNameAndLogo__oKv+O]");
            product.Name = nameNode.InnerText;
            product.Url = itemUrls[i];
            HtmlNode imageNode =
                doc.DocumentNode.SelectSingleNode("//picture[@class=Image__cardImage__xvgs1]");
            product.ImageUrl = imageNode.ChildNodes[1].Attributes["src"].Value;

            products.Add(product);
        }
        catch (Exception ex)
        {
            Console.WriteLine(ex.Message);
        }
        if (progress != null)
        {
            progress.Report((int)Math.Round((i + 1) / (double)count * 100));
        }
    }
}
return products;
}
```

Код Program

```

// See https://aka.ms/new-console-template for more information
using AnswearParser;
using AnswearParser.Models;
using HtmlAgilityPack;
using System;

Console.WriteLine("Program Start!");
var parser = new Parser();
//var products = parser.ParseWithParams("Women",1);
//var products = parser.ParseFromUrl("https://answear.ua/m/karl-lagerfeld", 1);
var progress = new Progress<int>(x=>x = 1);
var products = await parser.ParseFromUrlAsync("https://answear.ua/m/karl-lagerfeld", 1, progress);
foreach (var product in products)
{
    Console.WriteLine(product.Name);
    Console.WriteLine(product.Price);
    Console.WriteLine(product.Description);
    Console.WriteLine(product.Url);
    Console.WriteLine(product.ImageUrl);
    Console.WriteLine("-----");
}
//List<string> urls = new List<string>();

//string url = "https://answear.ua/k/vin/odyag";
//HtmlWeb web = new HtmlWeb();
//HtmlDocument doc = web.Load(url);

//foreach (HtmlNode productNode in doc.DocumentNode.SelectNodes("//div[@class='ProductItem__productCardImageWrapper__8Is-g']/a"))
//{
//    Console.WriteLine(productNode);
//    urls.Add(productNode.Attributes["href"].Value);
//    // Обробляємо інформацію про товар
//}
//var products = ParseInfoFromList(urls);
//List<Product> ParseInfoFromList(List<string> itemUrls)
//{
//    List<Product> products = new List<Product>();

//    foreach (string url in itemUrls)
//    {
//        HtmlDocument doc = web.Load("https://answear.ua" + url);
//        HtmlNode priceNode = doc.DocumentNode.SelectSingleNode("//div[@class='Price__price__CbfdW
ProductCard__priceRegular__sFepg']");
//        Product product = new Product();
//        product.Price = priceNode.InnerText;
//        HtmlNode descriptionNode =
doc.DocumentNode.SelectSingleNode("//div[@class='Accordion__accordionDescription__CckLI']");
//        var description = descriptionNode.ChildNodes.Select(x=>x.InnerText);
//        product.Description = string.Join(";", description);
//        HtmlNode nameNode =
doc.DocumentNode.SelectSingleNode("//div[@class='ProductCard__productNameAndLogo__oKv+O']");
//        product.Name = nameNode.InnerText;
//        product.Url = url;
//        HtmlNode imageNode = doc.DocumentNode.SelectSingleNode("//picture[@class='Image__cardImage__xvgs1']");
//        product.ImageUrl = imageNode.ChildNodes[2].Attributes["src"].Value;
//        products.Add(product);
//    }
//    return products;
//}

Console.WriteLine("Program Stop!");

```


Код DataContext

```

using AnswearParser.Models;
using Microsoft.EntityFrameworkCore;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Controls;

namespace ParserClient.Data
{
    class DataContext : DbContext
    {
        public DbSet<Product> Products { get; set; }

        protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
        {
            //Server = BORMAX\\SQLEXPRESS; Database = Restaurant; Trusted_Connection = True; Encrypt = False
            optionsBuilder.UseSqlServer("Server = (localdb)\\mssqllocaldb; Database = Answear; Trusted_Connection =
            True; MultipleActiveResultSets = true");
        }

        public DataContext()
        {
            //Database.EnsureDeleted();
            Database.EnsureCreated();
        }

        //protected override void OnModelCreating(ModelBuilder modelBuilder)
        //{
        //    modelBuilder.Entity<DishIngredient>()
        //        .HasKey(x => x.Id);
        //}
    }
}

```

Код FileSystem

```
using AnswearParser.Models;
using System;
using System.Collections.Generic;
using System.Collections.ObjectModel;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Xml.Serialization;

namespace ParserClient.Models
{
    public class FileSystem
    {
        public static void SaveToFile(List<Product> Products)
        {
            var dialog = new Microsoft.Win32.SaveFileDialog();
            dialog.Filter = "XML files (*.xml)*.xml|All files (*.*)*.*";
            if (dialog.ShowDialog() == true)
            {
                string filePath = dialog.FileName;
                // збереження даних файлу по шляху filePath
                var serializer = new XmlSerializer(typeof(List<Product>));
                using (var writer = new StreamWriter(filePath))
                {
                    serializer.Serialize(writer, Products);
                }
            }
        }
    }
}
```


Код RelayCommand

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Input;

namespace ParserClient.Models
{
    public class RelayCommand : ICommand
    {
        private Action<object> _execute;
        private Func<object, bool> _canExecute;

        public event EventHandler CanExecuteChanged
        {
            add { CommandManager.RequerySuggested += value; }
            remove { CommandManager.RequerySuggested -= value; }
        }

        public RelayCommand(Action<object> execute, Func<object, bool> canExecute = null)
        {
            _execute = execute;
            _canExecute = canExecute;
        }

        public bool CanExecute(object parameter)
        {
            return _canExecute == null || _canExecute(parameter);
        }

        public void Execute(object parameter)
        {
            _execute(parameter);
        }
    }
}
```

Код MainPageVM

```

using AnswerParser;
using AnswerParser.Models;
using ParserClient.Data;
using ParserClient.Models;
using System;
using System.Collections.Generic;
using System.Collections.ObjectModel;
using System.ComponentModel;
using System.Diagnostics;
using System.Linq;
using System.Runtime.CompilerServices;
using System.Security.Policy;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;

namespace ParserClient.ViewModels
{
    class MainPageVM : ViewModelBase, IPageViewModel
    {
        public MainPageVM()
        {
            PageCount = 1;
            Url = "https://answer.ua/k/vin/odyag";
            Products = new ObservableCollection<Product>();
        }

        public string Name => "CreditCardSearchOutline";

        private RelayCommand _parseCommand;
        private int _progress;

        public int Progress
        {
            get => _progress;
            set
            {
                _progress = value;
                OnPropertyChanged();
            }
        }

        private ObservableCollection<Product> _products;

        public ObservableCollection<Product> Products
        {
            get => _products;
            set
            {
                _products = value;
                OnPropertyChanged();
            }
        }

        private string _url;
        public string Url
        {
            get => _url;
            set
            {
                _url = value;
                OnPropertyChanged();
            }
        }

        private int _pageCount;

        public int PageCount
        {
            get => _pageCount;
            set
            {
                _pageCount = value;
                OnPropertyChanged();
            }
        }

        private TimeSpan _executionTime;

        public string ExecutionTime
        {
            get
            {
                if (_executionTime == TimeSpan.Zero)
                {
                    return string.Empty;
                }
            }
        }
    }
}

```


Продовження додатку К

```
    }
    else if (_executionTime.TotalSeconds < 1)
    {
        return "< 1 second";
    }
    else
    {
        return _executionTime.ToString(@"mm\:ss\.ff");
    }
}
set
{
    TimeSpan time;
    if (TimeSpan.TryParse(value, out time))
    {
        _executionTime = time;
        OnPropertyChanged();
    }
}
}

public RelayCommand ParseCommand
{
    get
    {
        return _parseCommand ??= new RelayCommand(async obj =>
        {
            if (String.IsNullOrEmpty(Url) || !Url.Contains("answer.ua"))
            {
                MessageBox.Show("Неправильні вхідні параметри");
                return;
            }
            var parser = new Parser();
            var stopwatch = new Stopwatch();
            stopwatch.Start(); // запуск таймера
            var progressIndicator = new Progress<int>(value => Progress = value);
            var products = await parser.ParseFromUrlAsync(Url, PageCount, progressIndicator);
            Products = new ObservableCollection<Product>(products);
            stopwatch.Stop(); // зупинка таймера
            ExecutionTime = stopwatch.Elapsed.ToString("hh\:mm\:ss");
            MessageBox.Show(ExecutionTime);
        });
    }
}

private RelayCommand _saveToFileCommand;
public RelayCommand SaveToFileCommand
{
    get
    {
        return _saveToFileCommand ??= new RelayCommand(async obj =>
        {
            if (Products.Count == 0)
            {
                MessageBox.Show("No products to save");
                return;
            }
            FileSystem.SaveToFile(new List<Product>(Products));
        });
    }
}

private RelayCommand _saveToDBCommand;
public RelayCommand SaveToDBCommand
{
    get
    {
        return _saveToDBCommand ??= new RelayCommand(async obj =>
        {
            if (Products.Count == 0)
            {
                MessageBox.Show("No products to save");
                return;
            }
            using (var db = new DataContext())
            {
                await db.Products.AddRangeAsync(Products);
                await db.SaveChangesAsync();
                MessageBox.Show("Успішно збережено");
            }
        });
    }
}
}
```

Код ParseByParamsVM

```

using AnswearParser;
using AnswearParser.Models;
using ParserClient.Data;
using ParserClient.Models;
using System;
using System.Collections.Generic;
using System.Collections.ObjectModel;
using System.Diagnostics;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;

namespace ParserClient.ViewModels
{
    public class ParseByParamsVM : ViewModelBase, IPageViewModel
    {
        public List<Category> Categories { get; set; }

        public List<Gender> Genders { get; set; }
        public string Name => "StoreSearchOutline";

        public ParseByParamsVM()
        {
            Categories = new List<Category>
            {
                new Category() { LogicalName = "odyag", Name = "Одяг" },
                new Category() { LogicalName = "vzutyta", Name = "Взуття" },
                new Category() { LogicalName = "aksesuary", Name = "Аксесуари" },
                new Category() { LogicalName = "odyag", Name = "Одяг" }
            };
            Genders = new List<Gender>
            {
                new Gender() { LogicalName = "Women", Name = "Жіночий" },
                new Gender() { LogicalName = "Men", Name = "Чоловічий" }
            };
            PageCount = 1;
            Products = new ObservableCollection<Product>();
        }

        private string _gender;
        public string Gender
        {
            get => _gender;
            set
            {
                _gender = value;
                OnPropertyChanged();
            }
        }

        private string _category;
        public string Category
        {
            get => _category;
            set
            {
                _category = value;
                OnPropertyChanged();
            }
        }

        private int _pageCount;
        public int PageCount
        {
            get => _pageCount;
            set
            {
                _pageCount = value;
                OnPropertyChanged();
            }
        }

        private ObservableCollection<Product> _products;
        public ObservableCollection<Product> Products
        {
            get => _products;
            set
            {
                _products = value;
                OnPropertyChanged();
            }
        }

        private TimeSpan _executionTime;
        public string Execution Time
        {
            get
            {

```


Продовження додатку Л

```
        if (_executionTime == TimeSpan.Zero)
        {
            return string.Empty;
        }
        else if (_executionTime.TotalSeconds < 1)
        {
            return "< 1 second";
        }
        else
        {
            return _executionTime.ToString(@"mm:ss.fff");
        }
    }
    set
    {
        TimeSpan time;
        if (TimeSpan.TryParse(value, out time))
        {
            _executionTime = time;
            OnPropertyChanged();
        }
    }
}
private int _progress;
public int Progress
{
    get => _progress;
    set
    {
        _progress = value;
        OnPropertyChanged();
    }
}
private RelayCommand _parseCommand;
public RelayCommand ParseCommand
{
    get
    {
        return _parseCommand ??= new RelayCommand(async obj =>
        {
            if (PageCount < 0 | String.IsNullOrEmpty(Gender) | String.IsNullOrEmpty(Category))
            {
                MessageBox.Show("Неправильні вхідні параметри");
                return;
            }
            var parser = new Parser();
            var stopwatch = new Stopwatch();
            stopwatch.Start(); // запуск таймера
            var progressIndicator = new Progress<int>(value => Progress = value);
            var products = await parser.ParseWithParamsAsync(Gender, progressIndicator, Category, PageCount);
            Products = new ObservableCollection<Product>(products);
            stopwatch.Stop(); // зупинка таймера
            ExecutionTime = stopwatch.Elapsed.ToString(@"hh\:mm\:ss");
            MessageBox.Show(ExecutionTime);
        });
    }
}
private RelayCommand _saveToFileCommand;
public RelayCommand SaveToFileCommand
{
    get
    {
        return _saveToFileCommand ??= new RelayCommand(async obj =>
        {
            if (Products.Count == 0)
            {
                MessageBox.Show("No products to save");
                return;
            }
            FileSystem.SaveToFile(new List<Product>(Products));
        });
    }
}
private RelayCommand _saveToDBCommand;
public RelayCommand SaveToDBCommand
{
    get
    {
        return _saveToDBCommand ??= new RelayCommand(async obj =>
        {
            if (Products.Count == 0)
            {
                MessageBox.Show("No products to save");
                return;
            }
            using (var db = new DataContext())
            {
                await db.Products.AddRangeAsync(Products);
                await db.SaveChangesAsync();
                MessageBox.Show("Успішно збережено");
            }
        });
    }
}
}
```

Код ViewModelBase

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Runtime.CompilerServices;
using System.Text;
using System.Threading.Tasks;

namespace ParserClient.ViewModels
{
    public class ViewModelBase : INotifyPropertyChanged
    {
        public event PropertyChangedEventHandler PropertyChanged;

        protected virtual bool SetProperty<T>(ref T storage, T value, [CallerMemberName] string propertyName = null)
        {
            if (Equals(storage, value)) return false;
            storage = value;
            OnPropertyChanged(propertyName);
            return true;
        }
        protected virtual void OnPropertyChanged([CallerMemberName] string propertyName = null)
        {
            var eventHandler = PropertyChanged;
            if (eventHandler != null)
                eventHandler(this, new PropertyChangedEventArgs(propertyName));
        }
    }
}
```


Код Gender

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;
```

```
namespace ParserClient.Models
```

```
{
```

```
    public class Gender
```

```
    {
```

```
        public string Name { get; set; }
```

```
        public string LogicalName { get; set; }
```

```
    }
```

```
}
```



Код Category

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;
```

```
namespace ParserClient.Models  
{  
    public class Category  
    {  
        public string Name { get; set; }  
        public string LogicalName { get; set; }  
    }  
}
```



Код ApplicationViewModel

```

using ParserClient.Models;
using ParserClient.View;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Input;

namespace ParserClient.ViewModels
{
    public class ApplicationViewModel : ViewModelBase
    {
        #region Fields
        private ICommand _changePageCommand;

        private IPageViewModel _currentPageViewModel;
        private List<IPageViewModel> _pageViewModels;
        #endregion

        #region Properties/Commands
        public ICommand ChangePageCommand
        {
            get
            {
                if (_changePageCommand == null)
                {
                    _changePageCommand = new RelayCommand(
                        p => ChangeViewModel((IPageViewModel)p),
                        p => p is IPageViewModel);
                }
                return _changePageCommand;
            }
        }

        public List<IPageViewModel> PageViewModels
        {
            get
            {
                if (_pageViewModels == null)
                    _pageViewModels = new List<IPageViewModel>();
                return _pageViewModels;
            }
        }

        public IPageViewModel CurrentPageViewModel
        {
            get
            {
                return _currentPageViewModel;
            }
            set
            {
                if (_currentPageViewModel != value)
                {
                    _currentPageViewModel = value;
                    OnPropertyChanged("CurrentPageViewModel");
                }
            }
        }
        #endregion

        #region methods
        private void ChangeViewModel(IPageViewModel viewModel)
        {
            if (!PageViewModels.Contains(viewModel))
                PageViewModels.Add(viewModel);

            CurrentPageViewModel = PageViewModels

```

Продовження додатку П

```
        .FirstOrDefault(vm => vm == viewModel);
        viewModel.Update();
    }
    public ApplicationViewModel()
    {
        // Add available pages
        PageViewModels.Add(new MainPageVM());
        PageViewModels.Add(new ParseByParamsVM());
        PageViewModels.Add(new ProductsViewModel());
        // Set starting page
        CurrentPageViewModel = PageViewModels[0];
    }
}
#endregion
```



Код IPageViewModel

```
using System;  
using System.Collections.Generic;  
using System.Diagnostics;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;
```

```
namespace ParserClient.ViewModels
```

```
{
```

```
    public interface IPageViewModel
```

```
    {
```

```
        string Name { get; }
```

```
        public void Update()
```

```
        {
```

```
            Debug.WriteLine("Update");
```

```
        }
```

```
    }
```

```
}
```



Код AssemblyInfo

using System.Windows;

[assembly: ThemeInfo(

ResourceDictionaryLocation.None, //where theme specific resource dictionaries are located

//(used if a resource is not found in the page,

// or application resource dictionaries)

ResourceDictionaryLocation.SourceAssembly //where the generic resource dictionary is located

//(used if a resource is

not found in the page,

// app, or any theme

specific resource dictionaries)

)]

