

# ВИПУСКНИЙ КВАЛІФІКАЦІЙНИЙ ПРОЄКТ

на тему:

## «Програмний модуль парсингу сайтів мереж продуктових магазинів»

Студента 4 курсу, 6 групи,  
спеціальності 121 «Інженерія  
програмного забезпечення»  
освітньої програми «Інженерія  
програмного забезпечення»

Шарапановського  
Данила Олеговича

\_\_\_\_\_

підпис студента

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

Власенко Лідія  
Олександрівна

\_\_\_\_\_

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

Шарапановський Данило Олегович

(прізвище, ім'я, по батькові)

1. Тема випускного кваліфікаційного проєкту «Програмний модуль парсингу сайтів мереж продуктових магазинів»

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 дослідити та створити програмний модуль парсингу сайтів  
мереж продуктових магазинів

Об'єкт дослідження процес парсингу інформації з сайтів

Предмет дослідження розробка програмного модулю парсингу



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 ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ

1.1. Характеристика роботи парсерів

1.2. Актуальність парсерів

1.3. Огляд існуючих рішень

1.4. Висновок до розділу 1

РОЗДІЛ 2. ВИБІР ПРОГРАМНИХ ЗАСОБІВ ТА ПРОЄКТУВАННЯ ПРОГРАМНОГО МОДУЛЮ

2.1. Вибір мови програмування

2.2. Вибір реляційної бази даних

2.3. Проєктування структури

2.4. Висновок до розділу 2

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПРОГРАМНОГО МОДУЛЮ

3.1. Розробка модулю парсера

3.2. Вирішення проблем з модулем

3.3. Висновок до розділу 3

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

ДОДАТКИ

## 6. Календарний план виконання проєкту

№ пор.	Назва етапів випускного кваліфікаційного проєкту	Строк виконання етапів проєкту	
		за планом	фактично
1	2	3	4
•	<i>Вибір теми випускного кваліфікаційного проєкту</i>	21.09.2022	21.09.2022
•	<i>Розробка та затвердження завдання на проєкт</i>	14.11.2022	14.11.2022
•	<i>Вступ та перелік літературних джерел</i>	23.12.2022	23.12.2022
•	<i>Розділ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ</i>	27.01.2023	27.01.2023
•	<i>Розділ 2. ВИБІР ПРОГРАМНИХ ЗАСОБІВ ТА ПРОЄКТУВАННЯ ПРОГРАМНОГО МОДУЛЮ</i>	03.03.2023	03.03.2023
•	<i>Розділ 3. РЕАЛІЗАЦІЯ ПРОГРАМНОГО МОДУЛЮ</i>	14.04.2023	14.04.2023
•	<i>Висновки</i>	28.04.2023	28.04.2023
•	<i>Здача випускного кваліфікаційного проєкту на кафедрі (перша перевірка)</i>	17.05.2023	17.05.2023
•	<i>Підготовка автореферату та презентації доповіді</i>	26.05.2023	26.05.2023
•	<i>Попередній захист випускного кваліфікаційного проєкту</i>	29.05.2023 – 02.06.2023	
•	<i>Зовнішнє рецензування випускного кваліфікаційного проєкту</i>	05.06.2023	05.06.2023
•	<i>Здача прошитого випускного кваліфікаційного проєкту на кафедрі</i>	05.06.2023	05.06.2023
•	<i>Публічний захист випускного кваліфікаційного проєкту</i>		

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

8. Науковий керівник випускного кваліфікаційного проєкту \_\_\_\_\_

Власенко Л.О.

(прізвище, ініціали, підпис)

9. Гарант освітньої програми \_\_\_\_\_

Рзаєва С.Л.

(прізвище, ініціали, підпис)

10. Завдання прийняв до виконання студент \_\_\_\_\_

Шарапановський Д.О.

(прізвище, ініціали, підпис)





## АНОТАЦІЯ

Випускний кваліфікаційний проєкт на тему: «Програмний модуль парсингу сайтів мереж продуктових магазинів».

Відповідно до мети дослідження робота присвячена створенню програмного модулю парсингу. В результаті порівняльного аналізу аналогічних рішень визначено їх функціонал та наявність певних переваг або недоліків.

Розроблений модуль парсингу має переваги над веб-сайтами з порівняннями цін та вже готовими рішеннями у вигляді програмного модуля.

Розроблений на мові програмування Python, в якості бази даних було використано MySQL

Досвід, набутий під час виконання досліджень та розробки, а також результати, які були отримані під час всієї роботи, можуть бути застосовані надалі при вирішенні схожих задач.

**Ключові слова:** парсинг, інструменти розробки, мова програмування, база даних, UML діаграми



## ABSTRACT

Graduation qualification project on the topic: "Software module for parsing websites of grocery store chains."

In accordance with the purpose of the research, the work is devoted to the creation of a software parsing module. As a result of a comparative analysis of similar solutions, their functionality and the presence of certain advantages or disadvantages were determined.

The developed parsing module has advantages over price comparison websites and ready-made solutions in the form of a software module.

Developed in the Python programming language, MySQL was used as a database

The experience gained during research and development, as well as the results obtained during the entire work, can be applied in the future when solving similar problems.

**Keywords:** parsing, development tools, programming language, database, UML diagrams

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ПЗ – програмне забезпечення

БД – база даних

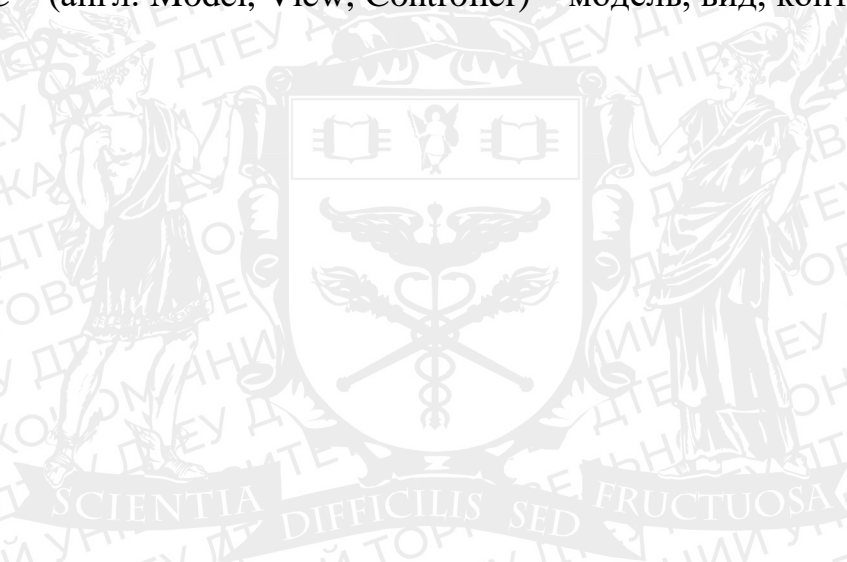
WEB – (англ. World Wide Web) – інтернет-простір

UML – (англ. Unified Modeling Language) - уніфікована мова моделювання

РРЦ – рекомендована роздрібна ціна

HTML – (англ. HyperText Markup Language) - мова розмітки гіпертексту

MVC – (англ. Model, View, Controller) – модель, вид, контролер



<i>ДТЕУ 121 06-26.БР</i>				
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докum.</i>	<i>Підпи</i>	<i>Дата</i>
Зав. каф.	Криворучко О.В.			14.04.23
Керівник	Власенко Л.О.			14.04.23
Гарант	Рзаєва С.Л.			14.04.23
Розробив	Шарапановський Д.О.			14.04.23
Програмний модуль парсингу сайтів мереж продуктових магазинів				
<i>Перелік умовних скорочень</i>				
<i>Стадія</i>			<i>Аркуш</i>	
<i>ПС</i>			<i>2</i>	
<i>Аркушів</i>			<i>50</i>	
<i>Факультет інформаційних технологій</i>				
<i>4 курс, 6 група</i>				



## ЗМІСТ

<b>ВСТУП.....</b>	<b>3</b>
<b>РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ .....</b>	<b>5</b>
1.1. Характеристика роботи парсингу .....	5
1.2. Актуальність парсерів .....	8
1.3. Огляд існуючих рішень .....	9
1.4. Висновки до розділу 1 .....	14
<b>РОЗДІЛ 2 ВИБІР ПРОГРАМНИХ ЗАСОБІВ ТА ПРОЄКТУВАННЯ</b>	
<b>ПРОГРАМНОГО МОДУЛЮ .....</b>	<b>15</b>
2.1. Вибір мови програмування .....	15
2.2. Вибір реляційної бази даних .....	18
2.3. Проєктування структури .....	21
2.4. Висновки до розділу 2 .....	23
<b>РОЗДІЛ 3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО МОДУЛЮ.....</b>	<b>24</b>
3.1. Розробка модулю парсера .....	24
3.2. Вирішення проблем з модулем .....	29
3.3. Висновок до розділу 3 .....	31
<b>ВИСНОВКИ ТА ПРОПОЗИЦІЇ.....</b>	<b>32</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....</b>	<b>33</b>
<b>КЕРІВНИЦТВО КОРИСТУВАЧА.....</b>	
<b>ДОДАТКИ.....</b>	

					<i>ДТЕУ 121 06-26.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпи</i>	<i>Дата</i>	Програмний модуль парсингу сайтів мереж продуктових магазинів  <i>Зміст</i>	<i>Стадія</i>	<i>Арку</i>	<i>Аркушів</i>
Зав. каф.		Криворучко О.В.		23.12.22		3	3	50
Керівник		Власенко Л.О.		23.12.22		<i>Факультет інформаційних технологій 4 курс, 6 група</i>		
Гарант		Рзаєва С.Л.		23.12.22				
Розробив		Шарапановський Д.О.		23.12.22				

## ВСТУП

*Актуальність.* Люди мають фундаментальну потребу в продуктах харчування. Тож проектування та створення парсеру для сайтів продуктових мереж є вкрай важливим. Тим паче, коли має місце кризова ситуація або війна. Завдяки парсеру, можна буде швидко збирати і аналізувати інформацію та порівнювати ціни на продукти харчування.

Продуктові мережі можуть оперативнo реагувати на зміни в умовах постачання та завжди надавати клієнтам повністю актуальну інформацію. А також парсер допоможе виявити дефіцит товарів у певних магазинах, це забезпечить рівномірне постачання товару. Чи допоможе проаналізувати асортимент продукції конкурентів.

Волонтери ж можуть зорієнтуватися, де найближчі магазини із найнижчими цінами, аби якомога ефективніше розпорядитися своїм бюджетом та збільшити кількість людей, яким зможуть допомогти.

Також і для звичайних людей користування парсеру стане в пригоді. Для багатьох сімей, а особливо в умовах економічних криз, ефективне планування бюджету є надзвичайно важливим. Вони зможуть знаходити знижки та спеціальні пропозиції, таким чином в сімейному бюджеті з'являться більше грошей для оплати інших послуг, наприклад, комунальних платежів.

Тому, це універсальний інструмент, який полегшує прийняття складних рішень та значно підвищує ефективність вибору під час закупівлі продуктів харчування.

					<i>ДТЕУ 121 06-26.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпи</i>	<i>Дата</i>				
Зав. каф.		Криворучко О.В.		23.12.22	Програмний модуль парсингу сайтів мереж продуктових магазинів	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Керівник		Власенко Л.О.		23.12.22		<i>В</i>	<i>4</i>	<i>50</i>
Гарант		Рзаєва С.Л.		23.12.22		<i>Факультет інформаційних технологій 4 курс, 6 група</i>		
Розробив		Шарапановський Д.О.		23.12.22				
					<i>Вступ</i>			



*Мета дослідження:* Дослідження та розробка програмного модуля парсинга сайтів мереж продуктових магазинів.

*Об'єкт дослідження:* процес парсингу інформації з сайтів.

*Предмет дослідження:* розробка програмного модулю парсингу.

У відповідності з метою дослідження поставлені наступні завдання:

- Розглянути та проаналізувати подібні рішення;
- Дослідити проблематики парсингу;
- Розглянути функціональності системи;
- Визначити структуру проєкту.

*Методи дослідження:* опис, аналіз, пояснення, моделювання.

*Наукова новизна дослідження* полягає в представленні розробленого програмного модуля парсингу, який відповідає всім критеріям.

*Практичне значення дослідження:* розроблений програмний модуль буде корисним інструментом для багатьох груп користувачів: бізнесів, волонтерів, або звичайних людей.

								Аркуш
								6
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 06-26.БР			

## РОЗДІЛ 1

### АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ

#### 1.1. Характеристика роботи парсингу

Парсери – це програмні компоненти, які приймають вхідні дані, зазвичай текст, створюють структури даних часто дерева розбору, дерева абстрактного синтаксису або інші ієрархічні структури і забезпечують структурне представлення вхідних даних, перевіряючи правильність синтаксису.

Парсинг цін – це процес сканування та вилучення цін із сайтів заздалегідь вибраних інтернет-магазинів з подальшою конвертацією даних у зручний для аналізу формат. [1]

Парсинг цін – один із основних інструментів комерційних та маркетингових відділів торгових онлайн майданчиків для формування власної цінової стратегії ціноутворення, а виробникам та постачальникам дає розуміння ринкових тенденцій та можливість контролю РРЦ. [1]

Вхідними даними для парсерів часто є текст якоїсь «комп'ютерної мови», але це також може бути звичайний текст або неструктуровані текстові дані, в цьому випадку зазвичай витягуються лише певні частини тексту, а не будується дерево синтаксичного аналізу.

Синтаксичні аналізатори можуть варіюватися від дуже простих функцій, до більш складних програм, таких як HTML-аналізатори веб-сторінок.

Парсери взаємодіють з ієрархічним деревом веб-сторінки(Рис.1.1)

Зм.	Аркуш	№ докум.	Підпи	Дата	ДТЕУ 121 06-26.БР			
Зав. каф.		Криворучко О.В.		27.01.23	Програмний модуль парсингу сайтів мереж продуктових магазинів	Стадія	Аркуш	Аркушиів
Керівник		Власенко Л.О.		27.01.23		РІ	7	50
Гарант		Рзаєва С.Л.		27.01.23	АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ	Факультет інформаційних технологій		
Розробив		Шарапановський Д.О.		27.01.23		4 курс, 6 група		



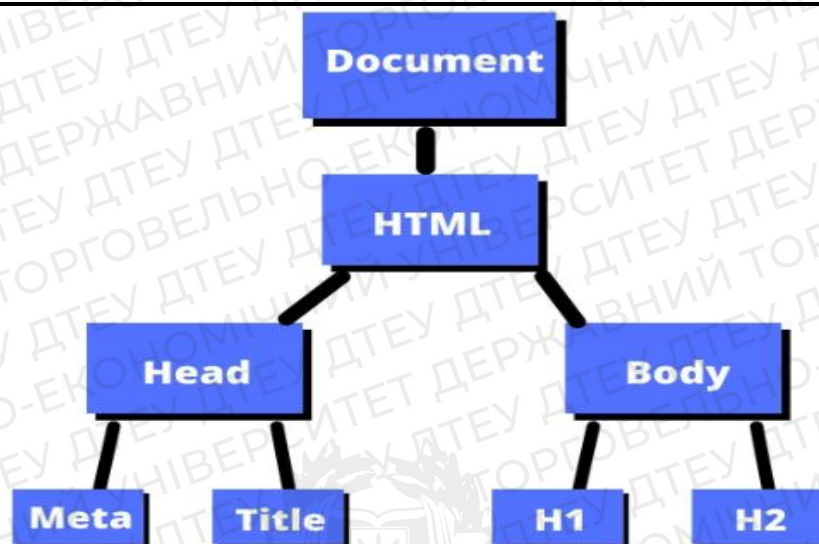


Рис. 1.1 Ієрархічне дерево веб-документа

для отримання потрібної інформації та для навігації по структурі самої сторінки.

Парсер має можливість переглядати кожен елемент, що знаходиться в ієрархічному дереві обраної веб-сторінки та отримувати інформацію про них, таку як:

- назва тега;
- атрибути;
- вміст.

Тобто він може зчитувати, наприклад, всі елементи певного типу, які мають інформацію про товари на сторінці продуктового магазину.

Однією з можливостей є пересування парсера по ієрархічному дереву веб-сторінок, використовуючи методи для переходу до наступного, попереднього, або ж батьківського елемента.

Парсери можуть маніпулювати структурою дерева. Може додавати, видаляти, або вносити зміни в елементи в ієрархічному дереві веб-сторінки.

					ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		8

Пошук за селекторами, парсери можуть використовувати селектори, наприклад, **Css-селектори** для точного вибору певних елементів в ієрархічному дереві веб-сторінок.

Також парсери можуть бути корисними використання парсера й власному сайті.

Це дозволить проводити аналітику власного сайту за допомогою парсера. Можна збирати та аналізувати дані з інших джерел для проведення порівняння.

Можливість знайти посилання, які були пошкоджені та запровадити автоматичне оновлення інформації для запобігання виникненню таких ситуацій у подальшому.

Кардинально покращити якість даних представлених на сайті. Адже парсер виявить неправильно введені дані. Це покращить досвід користуванням сайту для його відвідувачів.

Значною мірою парсер можна назвати універсальним та корисним інструментом для якого знайдеться велике коло застосування.

Вони дозволяють витягувати необхідну інформацію зі сторінок та використовувати її для різних цілей. Завдяки широким можливостям парсерів, їх використання стає дедалі більш поширеним у різних галузях, де потрібна автоматизація та обробка веб-даних.

									ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата						8



## 1.2. Актуальність парсерів

Під час активного зростання масиву інформації на різноманітних інтернет ресурсах виникає потреба в ефективних рішеннях обробки цієї самої інформації. Тому на допомогу приходять парсер.

Використання парсинга є актуальним рішенням в таких випадках:

- Отримання даних з великої кількості веб-сторінок;
- Оновлення інформації на власному сайті в автоматичному режимі;
- Вибір критерій за якими відбудеться пошук інформації;
- Поєднання інформації з різних ресурсів;
- Збереження даних в зручному для користувача вигляді;

Також він має такі переваги над іншими видами збору інформації:

1. Точність – не буде «людського фактору» (у вигляді помилок) при пошуці інформації;
2. Економія часу – час виконання завдання зменшиться в декілька разів
3. Автоматизація – можливість в автономному режимі збирати потрібну інформацію

Вже після використання парсера інформація, котра була вибрана користувачем, в зручному вигляді може бути збережена в CSV форматі або в базі даних. У подальшому може бути легко відредагована, доповнена, а потім за потреби використовуватися.

						ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			8

### 1.3. Огляд існуючих рішень

Тепер розглянемо існуючі рішення, одним з варіантів є вебресурси, які надають інформацію по обраному товару.



Рис. 1.2 Знімок інтерфейсу сайту GoToShop

На сайті GoToShop за пошуком товару можемо знайти таку інформацію:

- Список магазинів з найнижчою ціною;
- Стислий опис товару;
- Характеристика (склад, вагу і тд.).

					ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		8





Рис. 1.3 Знімок інтерфейсу сайту SuperMarkets

Сайт з моніторингу цін SuperMarkets надає схожий функціонал, що і сайт GoToShop, а саме:

- Список магазинів з найнижчою ціною;
- Стислий опис товару;
- Характеристика (склад, вагу і тд.).

Також декілька нових функцій:

- Рейтинг товару ;
- Мапа з магазинами, які мають товар.

						ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			8

## Порівняння вебресурсів моніторингу цін

Назва	GoToShop	SuperMarkets
Дизайн	+	+
Швидкодія	+	+
Інформативність	+	+
Інтуативність	+	+
Актуальність інформації	+	-
Можливість відслідковування декількох товарів	-	-

Після порівняння цих сайтів можна зробити висновок, що не всі подібні ресурси можуть надавати актуальну інформацію. Адже на другому сайті не було актуальної пропозиції з найнижчою ціною.

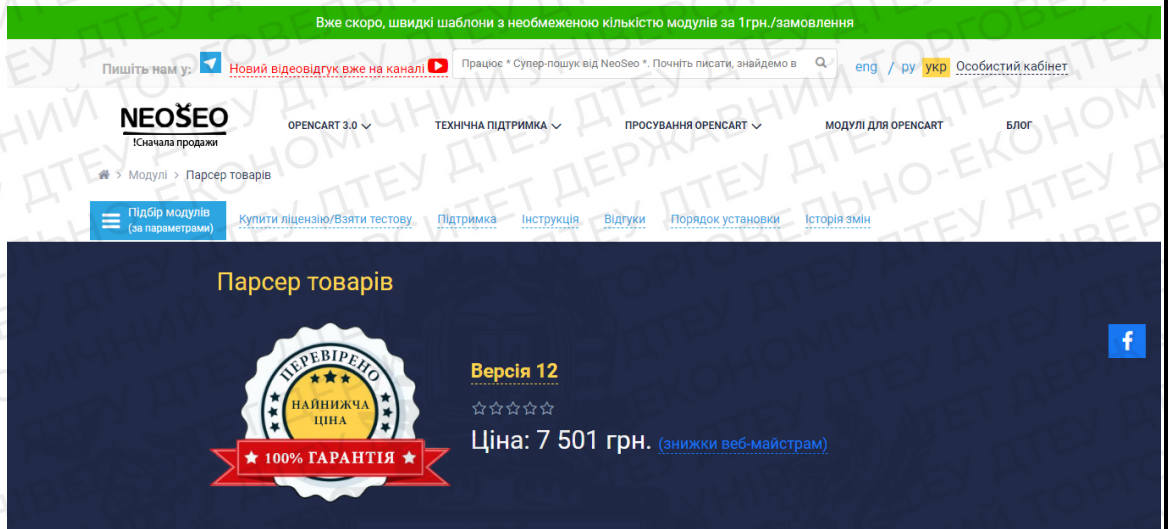
Використання такого типу сайтів є також малоефективним:

1. Моніторинг можна проводити тільки за одним товаром;
2. Сама перевірка товарів проводиться вручну;
3. Велика витрата часу для великих об'ємів;
4. Немає можливості автоматизувати процес.

					ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		8



Також існує альтернатива у вигляді програмного модулю парсингу. Є також можливість придбати вже готовий модуль, але в такому рішенні є свої недоліки, які далі будуть розглянуті.



### Інформація про модуль

Відразу після оформлення замовлення Ви зможете завантажити ліцензію і файли модуля в Особистому кабінеті

Своєчасне оновлення асортименту дозволяє утримувати увагу покупців і залучати нових відвідувачів. Але наповнення сайту вручну віднімає дуже багато часу, а також істотно уповільнює процес оновлення каталогу магазину. Але є вихід - модуль парсинга товарів з іншого сайту. Функціонал даного модуля має на увазі автоматичний пошук необхідної інформації про товар, використовуючи різні сайти конкурентів. Ви заощадите не тільки час, а й гроші на контент-відділі. Особливо актуальність даного модуля відчувається в передсвяткові дні, коли з'являється необхідність завантажити велику кількість товару з мінімальними тимчасовими витратами.

Розширюйте асортимент свого Інтернет-магазину швидко і легко!

Рис. 1.4 Знімок сайту з готовим модулем

Цей варіант має певні нюанси, а саме:

- Захмарна ціна готового програмного модулю;
- Обмежений функціонал;
- Проблеми з подальшою підтримкою.

								Аркуш
								8
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 06-26.БР			

## Порівняння вебресурсів з програмним модулем парсингу

Назва	Вебресурси з моніторингу цін	Програмний модуль парсингу
Моніторинг за групою товарів	-	+
Автоматизація	-	+
Витрата часу	велика	низька
Можливість кастомізації	-	+

З таблиці 1.2 можемо зробити висновок, проте що програмний модуль парсингу однозначно кращий вибір за вебресурси з моніторингу цін. Користуватися останнім доречно при здійсненні невеликих побутових закупівель. Однак програмний модуль підходить для вирішення великого кола завдань, а також буде корисним не тільки для рядового користувача, але і для бізнесів, мереж продуктових магазинів.

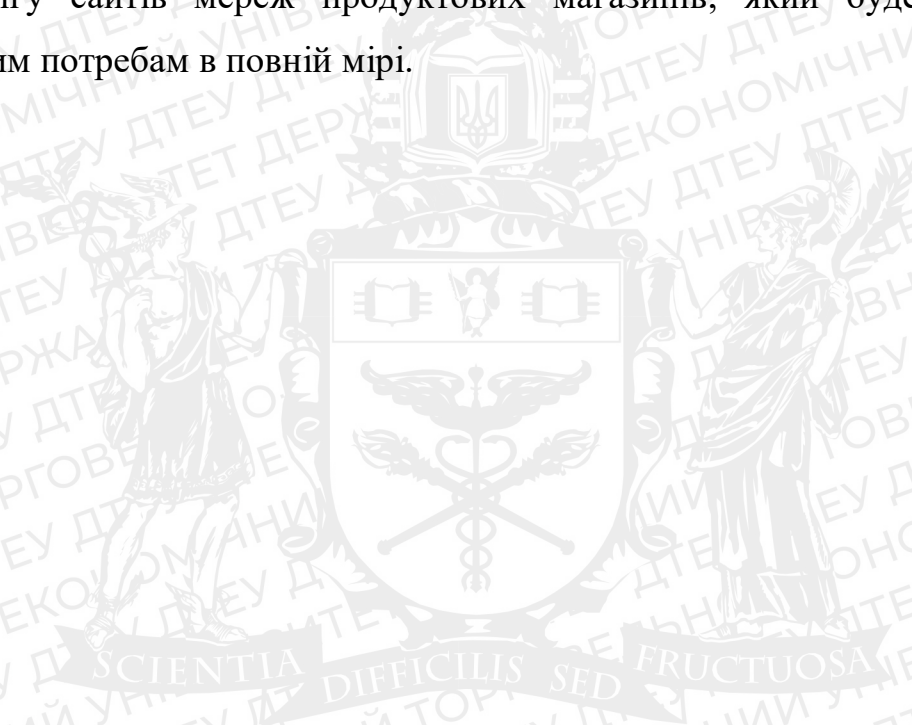
Проте після озвучених недоліків готового модулю, виникає потреба в більш гнучкому та менш затратному рішенні. Тому єдиним варіантом залишається розробити власний програмний модуль парсингу сайтів мереж продуктових магазинів.

					ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		8



#### 1.4. Висновки до розділу 1

Отже, в цьому розділі було розглянуто основи парсингу. Розібрана актуальність використання парсерів. Було проведено огляд існуючих рішень, у вигляді вебресурсів з моніторингу цін і вже програмного модуля, та порівняння їх за декількома критеріями. Підсумовуючи все вище викладене, можна визначити, що доцільно розробити власний програмний модуль парсингу сайтів мереж продуктових магазинів, який буде відповідати власним потребам в повній мірі.



					ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		8

## РОЗДІЛ 2

### ВИБІР ПРОГРАМНИХ ЗАСОБІВ ТА ПРОЄКТУВАННЯ ПРОГРАМНОГО МОДУЛЮ

#### 2.1. Вибір мови програмування

Виникає питання, яка мова програмування підходить для написання програмного модулю парсингу. Існує доволі багато мов програмування, які можна використати. Наприклад:

1. Python
2. JavaScript
3. Java
4. Ruby

В кожній з цих мов є свої сильні та слабкі сторони.

**JavaScript** є доволі високо поширеною, це означає, що можна знайти велику кількість документації. Має підтримку веб браузерів, бо Javascript вони виконують нативно, що зменшить потребу у встановленні додаткових інструментів.

Але також має великі недоліки. Javascript виконується на клієнтській стороні, що може призводити до обмеженої продуктивності. А також має строгу політику, що обмежує можливості цієї мови при парсингу інших сторінок.

**Java** також є дуже поширеною мовою, має доволі потужну типизацію, дозволить вчасно виявляти помилки. Java є досить потужною мовою, яка має високу продуктивність, має кросплатформу.

З недоліків мови програмування Java є складний синтаксис, який

					<i>ДТЕУ 121 06-26.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпи</i>	<i>Дата</i>				
Зав. каф.		Криворучко О.В.		03.03.23	Програмний модуль парсингу сайтів мереж продуктових магазинів  <i>Вибір програмних засобів та проєктування програмного модулю</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Керівник		Власенко Л.О.		03.03.23		P2	10	50
Гарант		Котенко Н.О.		03.03.23		<i>Факультет інформаційних технологій 4 курс, 6 група</i>		
Розробив		Шарапановський Д.О.		03.03.23				



збільшить витрату часу та буде забирати багато зусиль. Не дивлячись на широке використання вибір бібліотек є доволі обмеженим для парсингу.

**Ruby** має гарний синтаксис, який дуже легко читати та доволі зрозумілий, що може пришвидшити розробку на цій мові програмування. Має широкий вибір готових бібліотек, які зекономлять багато часу. Та взагалі має високу швидкість розробки.

Проте і ця мова має свої недоліки, а саме має гіршу швидкодію у порівнянні з тим же Python. Ruby вимагає більшу кількість ресурсів персонального комп'ютера, а саме оперативної пам'яті, що буде критичною проблемою при обробці великих об'ємів даних. Також варто зазначити низьку популярність з іншими мовами, що теж може сприяти виникненню певних пролем при розробці.

Проаналізувавши наведені вище мови програмування, вибір мови для розробки програмного модулю впав саме на Python.

**Python** – це одна з найпопулярніших і найпоширеніших мов програмування для розробки програмних модулів парсингу. Див рис. 2.1

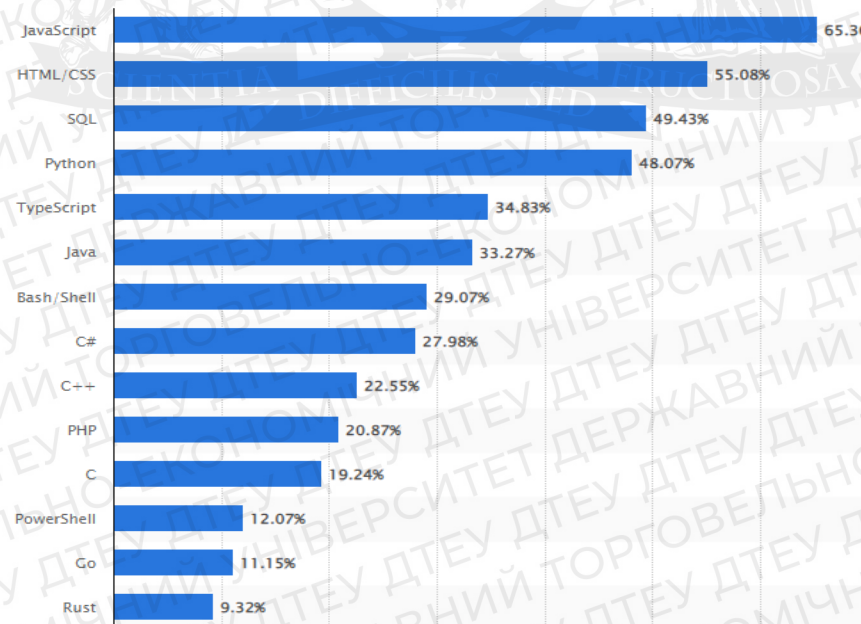


Рис. 2.1 графік популярності мов програмування [2]

Python має простий і зрозумілий синтаксис, що полегшує розробку і розуміння коду. Це робить Python майже ідеальним вибором для розробки парсерів, особливо для початківців у програмуванні.

Ця мова має обширну кількість вбудованих функцій і бібліотек, які будуть корисними для розробки парсеру і роботи з веб-сторінками, такі як: BeautifulSoup, Requests, Scrapy та інші. Ці бібліотеки спеціалізуються на аналізі html-сторінки та вилученні потрібних даних з них.

Також однією з вагомих переваг мови програмування Python є наявність великої спільноти розробників і фанатів. Тож легко можна знайти потрібну документацію, а також є велика кількість форумів, на яких буде легко знайти відповіді на свої питання.

З мінусів Python має високу залежність від зовнішніх бібліотек. Також має відсутність підтримки розробки для мобільних пристроїв: хоча Python й пропонує можливості крос-платформи, він не має надійної вбудованої підтримки для розробки мобільних додатків.

Хоча й існують такі фреймворки, як Kivy. Вони існують для створення мобільних додатків на Python, вони можуть не запропонувати такий самий рівень продуктивності та вбудованої інтеграції, як специфічні для платформи мови, такі як Swift.

Незважаючи на ці проблеми, Python залишається універсальною та дуже широко поширеною мовою програмування завдяки своїм великим бібліотекам, сильній підтримці спільноти та простоті використання.

Багато з цих проблем можна пом'якшити за допомогою належних методів кодування або методів оптимізації продуктивності та використання великої екосистеми Python.

						ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			17



## 2.2. Вибір реляційної бази даних

База даних являє собою структуровану сукупність даних. Ці дані можуть бути будь-якими - від простого списку майбутніх покупок до переліку експонатів картинної галереї або величезної кількості інформації в корпоративній мережі. [7]

Для запису, вибірки і обробки даних, що зберігаються в комп'ютерній базі даних, необхідна система управління базою даних.[7]

Для виконання поставленої мети є доволі великий вибір серед баз даних, наприклад:

1. SQLite
2. PostgreSQL
3. MySQL

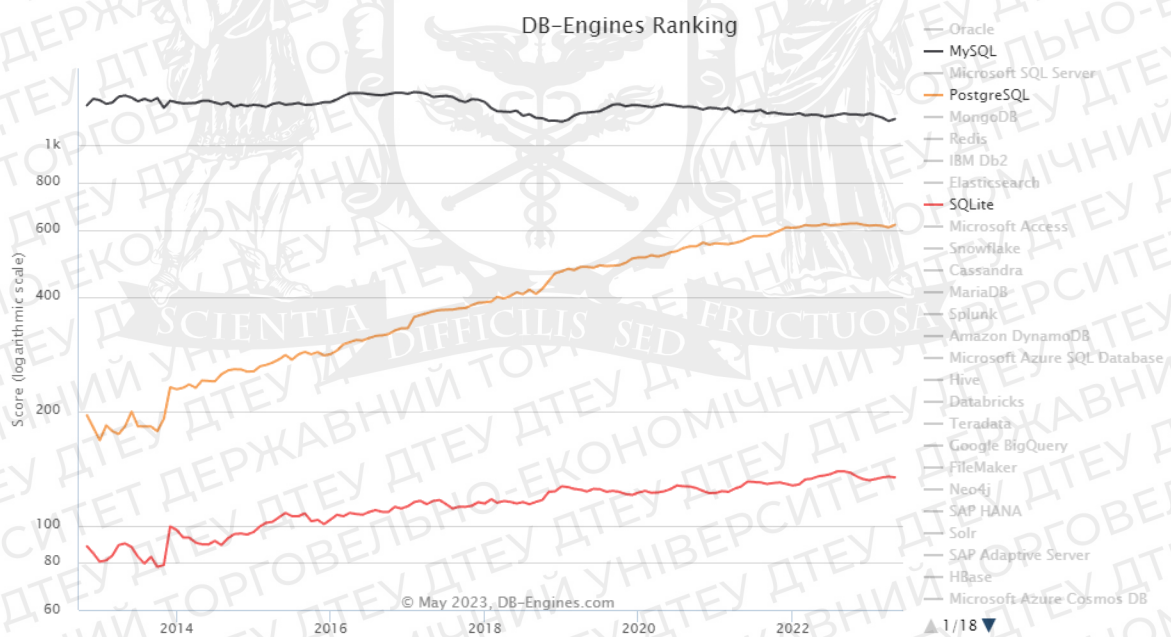


Рис.2.2 тенденція популярності БД [3]

Цей рисунок показує рівень популярності вказаних систем, з яких безперечним лідером була і залишається саме MySQL.

Проте варто розглянути й інші бази даних.

					Аркуш
					18
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 06-26.БР

**SQLite** доволі легка у використанні, має просту установку. Є доволі компактною, все зберігається в одному файлі. Не потребує встановлення та налаштування сервера, що зменшить час розгортання бази для невеликих проєктів.

Проте має вагомі недоліки у вигляді не підтримку багатопоточності, що звісно буде обмежувати продуктивність в проєктах з великими об'ємами даних. Має відчутну обмеженість функціоналу та деяких функцій, що може бути проблемним для складних проєктів. А ще SQLite не має мережевого доступу, тому що розрахована на локальне користування.

**PostgreSQL** похвалитися широкими можливостями, адже є потужною реляційною базою даних, з підтримкою розширень SQL, різноманітних типів даних. Має велику масштабованість, що дозволить працювати з великими об'ємами даних. Варто зазначити і безпеку цієї БД, а також можливість резервного копіювання та відновлення даних.

Однак, великі можливості ускладнюють налаштування та управління базою даних. Відчутним недоліком буде і те, що PostgreSQL вимагає достатньо багато ресурсів системи, а особливо оперативної пам'яті. Що може призводити до здорошення користування цією базою даних

Тому вибір був зроблений на користь бази даних – MySQL.

MySQL – це програмне забезпечення, яке має відкритий код, тобто будь-хто може його модифікувати. Воно повністю безкоштовне та займає просто лідируючі позиції, що можна побачити на Рис. 2.2.

Також з переваг відзначають:

- Надійність
- Простоту
- Плавну роботу

						ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			19



415 systems in ranking, May 2023

Rank			DBMS	Database Model	Score		
May 2023	Apr 2023	May 2022			May 2023	Apr 2023	May 2022
1.	1.	1.	Oracle +	Relational, Multi-model	1232.64	+4.36	-30.18
2.	2.	2.	MySQL +	Relational, Multi-model	1172.46	+14.68	-29.64
3.	3.	3.	Microsoft SQL Server +	Relational, Multi-model	920.09	+1.57	-21.11
4.	4.	4.	PostgreSQL +	Relational, Multi-model	617.90	+9.49	+2.61
5.	5.	5.	MongoDB +	Document, Multi-model	436.61	-5.29	-41.63
6.	6.	6.	Redis +	Key-value, Multi-model	168.13	-5.42	-10.89
7.	7.	7.	IBM Db2	Relational, Multi-model	143.02	-2.48	-17.31
8.	8.	8.	Elasticsearch	Search engine, Multi-model	141.63	+0.56	-16.06
9.	9.	↑ 10.	SQLite +	Relational	133.86	-0.68	-0.87
10.	10.	↓ 9.	Microsoft Access	Relational	131.17	-0.20	-12.27

Рис. 2.3 Рангова таблиця БД [3]

На рисунку 2.3 також видно на скільки балів MySQL випереджає більшість баз даних.

Величезною перевагою цієї БД в її вимогах до апаратного забезпечення:

З мінімальними вимогами

- 1 комп'ютер з процесором Pentium II 233 МГц і вище
- 128 Мб оперативної пам'яті

Та рекомендованими вимогами

- 2 комп'ютери з процесором Pentium II 233 МГц і вище
- 256Мб оперативної пам'яті

Також варто відзначити систему безпеки, яка Система безпеки заснована на привілеї та паролі з можливістю верифікації з віддаленого комп'ютера, за рахунок чого забезпечується гнучкість і безпеку. [7]

					ДТЕУ 121 06-26.БР	Аркуш
						20
Зм.	Аркуш	№ докум	Підпис	Дата		

### 2.3. Проектування структури

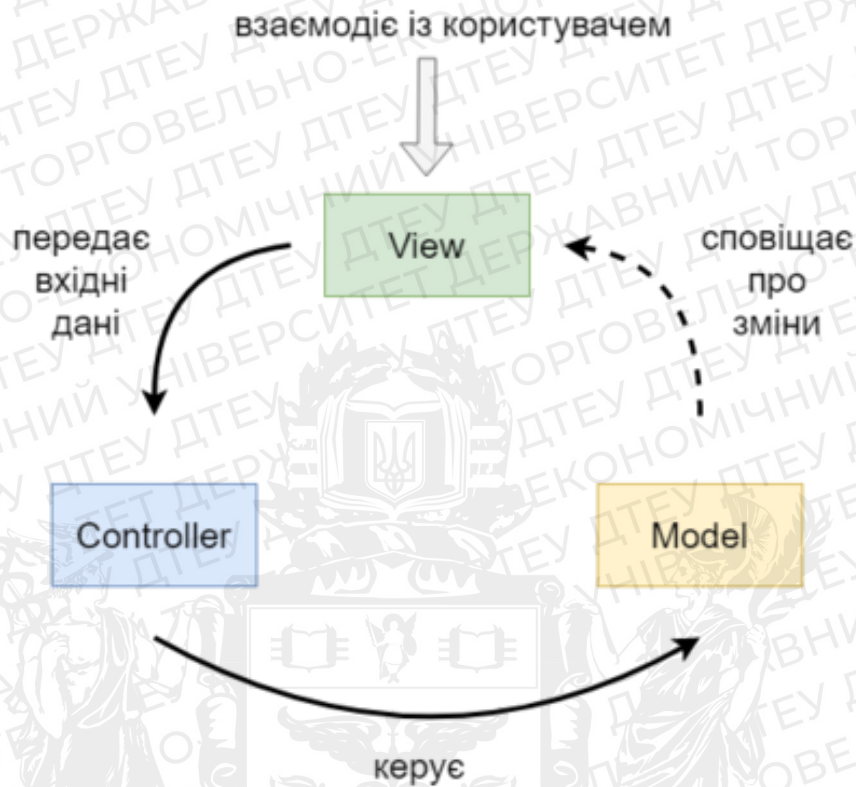


Рис. 2.4 Шаблон MVC

Цей шаблон MVC (Model, View, Controller), який описує роботу побудови структури додатку. Головною метою якого є розділення бізнес-логіки від зазначеного користувача інтерфейсу. Це означає, те що в майбутньому значно полегшить його масштабування.

В цій схемі, а саме Модель, Вигляд, Контролер передбачається поділ системи на три основні та взаємопов'язані частини: модель даних, інтерфейс користувача та модуль керування.

Шаблон використовується для відокремлення даних від інтерфейсу користувача, мінімізації впливу змін користувацького інтерфейсу на дані, а також для внесення змін до моделі даних без зміни для інтерфейсу користувача.

						ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			21



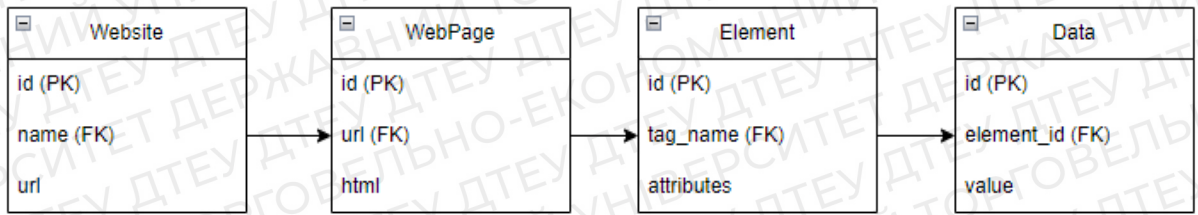


Рис. 2.5 Схема бази даних

Ця діаграма бази даних представляє спрощену схему для програми парсеру веб-сторінки.

Вона складається з чотирьох таблиць з назвами:

- Website,
- Webpage,
- Element,
- Data.

У Website таблиці зберігається інформація про веб-сайти, які потрібно розібрати. Він містить такі атрибути, як id, name і url.

Таблиця WebPage представляє окремі веб-сторінки в межах веб-сайту. Він містить такі атрибути, як id, url, html.

Таблиця Element представляє елементи HTML на веб-сторінці. Він має такі атрибути, як id, tag\_name, attributes.

Таблиця Data зберігає дані, отримані з елементів HTML. Він включає такі атрибути, як id, element\_id, value.

## 2.4. Висновки до розділу 2

Отже, підсумовуючи все вище викладене, в цьому розділі було розглянуто різні мови програмування з наведенням сильних і слабких сторін кожної з них. Було обрано Python для написання модулю парсера. Ще було обрано базу даних MySQL та знайдено оптимальні рішення для програмного модуля. А також було здійснено проектування з наведенням діаграм та описом до них.



								Аркуш
								23
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 06-26.БР			



## РОЗДІЛ 3

### РЕАЛІЗАЦІЯ ПРОГРАМНОГО МОДУЛЮ

#### 3.1. Розробка модулю парсера

Мовою програмування для розробки програмного модулю було обрано Python, а в якості бази даних було обрано MySQL.

Розробка буде проходити у середовищі розробки - PyCharm

При розробці будуть використовуватися бібліотеки beautiful soup, requests, time, mysql-connector-python.

В даному розділі буде описана розробка програмного модулю парсингу сайтів мереж продуктових магазинів.

Будуть наведені рисунки та частини програмного коду з їх поясненням, а також знаходження проблем та їх варіанти вирішення.

Вимоги до апаратної частини:

- Персональний комп'ютер на базі процесору AMD або Intel;
- Оперативної пам'яті не менше 1 гб;
- Місця на диску не менше 1 гб.
- Операційна система Windows

Зм.	Аркуш	№ докум.	Підпи	Дата	<i>ДТЕУ 121 06-26.БР</i>			
Зав. каф.		Криворучко О.В.		14.04.23	Програмний модуль парсингу сайтів мереж продуктових магазинів	Стадія	Аркуш	Аркушів
Керівник		Власенко Л.О.		14.04.23		РЗ	24	50
Гарант		Рзаєва С.Л.		14.04.23		Факультет інформаційних технологій		
Розробив		Шарапановський Д.О.		14.04.23		4 курс, 6 група		
					<i>Реалізація програмного модулю</i>			

```

    <div id="app" class="app"> flex
      <div class="menu-header-wrapper"> ... </div>
      <div class="l-body" id="l-body"> flex
        <aside class="menu-sidebar-wrapper"> ... </aside> flex
        <div class="l-main"> flex
          <div class="l-content">
            <div class="page-generator">
              <div class="hold-spacer">
                <ul class="bread-crumbs hold-spacer"> ... </ul>
                <div class="hold-spacer">
                  <div class="product-preview"> flex
                    <div class="product-block-wrapper"> ... </div>
                    <div class="labels"> ... </div> flex
                    <div class="product-preview__image"> ... </div>
                    <div class="product-preview-content">
                      <div class="product-preview__image-mobile"> ... </div>
                      <h1 class="product-preview__title"> ... </h1>
                      <div class="product-preview__weight"> ... </div>
                      <div class="product-preview__description"> ... </div>
                      <div class="product-preview__price-wrapper"> flex
                        <div class="product-price product-preview__price">
                          <div class="product-price__integer">29</div> == $0
                          <div class="product-price__other"> ... </div>
                        </div>

```

Рис 3.1 приклад коду html сторінки для парсингу

Для початку потрібно отримати html код веб-сторінки (для прикладу рис. 3.1). Його переглядаємо через інструменти розробника Google Chrome, для їх відкриття потрібно вибрати View page source або Inspect.

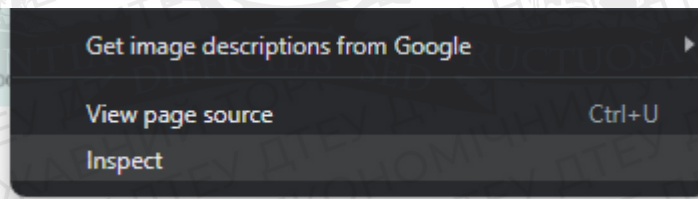


Рис. 3.2. перехід до інструментів розробника Chrome

І для цього використаємо вже згадану раніше бібліотеку requests.

Потрібно обрати веб-сайт з якого буде відбуватися парсинг інформації. Після чого потрібно скопіювати посилання на цей сайт і приступити до написання коду програмною мовою Python (рис.3.2).



```

1 import requests
2
3 # посилання на сайт для парсингу
4 url = "https://silpo.ua/"
5
6 response = requests.get(url)
7
8 if response.status_code == 200:
9     html_code = response.content
10
11 else:
12     print("Помилка при отриманні HTML-коду:", response.status_code)
13
14

```

Рис 3.3 програмний код requests

За допомогою бібліотеки requests ми отримаємо сторінку та зможемо продовжити процес парсингу. У разі виникнення проблем з доступом до сайту, отримаємо повідомлення « Помилка при отриманні HTML – коду: ».

Помилки з доступом до сайту можуть бути на стороні хосту самого сайту і в такому випадку користувач не зможе здійснити парсинг, проте якщо помилки будуть зі сторони користувача, можна запобігти цьому і способи як це зробити будуть розглянуті далі.

						ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			26

```

from bs4 import BeautifulSoup

soup = BeautifulSoup(html_code, "html.parser")

h1_tags = soup.find_all("h1")

# Виведемо назву товару і його ціну
for h1_tag in h1_tags:
    title = h1_tag.text.strip() # Отримуємо текст заголовка, видаляємо зайві пробіли
    price_tag = h1_tag.find_next_sibling("span", class_="price")
    # Знаходимо наступний сусідній елемент <span> з класом "price"

    if price_tag:
        price = price_tag.text.strip() # Отримуємо текст ціни і видаляємо зайві пробіли
        print("Назва товару:", title)
        print("Ціна:", price)
        print("-----")

```

Рис. 3.4 приклад коду парсера

Знаходимо текст заголовка `<h1>` і видаляємо зайві пробіли з використанням методу `strip()`. Отриману назву товару зберігаємо у змінну `title`.

Знаходимо наступний сусідній елемент `<span>` з класом `"price"` відносно поточного елемента `<h1>`. В результаті отримуємо елемент, що містить ціну товару.

Перевіряємо, чи знайдено елемент з ціною

Отримуємо текст ціни з елемента `<span>` і видаляємо зайві пробіли. Отриману ціну зберігаємо у змінну `price`.

						ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			27



```

32     import mysql.connector
33
34     def insert_data(product_name, price, description):
35         cursor = connection.cursor()
36         sql = "INSERT INTO products (product_name, price, description) VALUES (%s, %s, %s)"
37         values = (product_name, price, description)
38         cursor.execute(sql, values)
39         connection.commit()
40         cursor.close()
41
42     insert_data(product_name, price, description)
43
44

```

Рис. 3.5 код для створення бд

Потрібно встановити та налаштувати базу даних MySQL.

Встановити бібліотеку для взаємодії з базою даних SQL. А саме бібліотеку mysql-connector-python.

Використовуємо mysql-connector-python для взаємодії з MySQL базою даних. Ми виконуємо INSERT-запит для вставки даних у таблицю products, передаючи значення параметрів через кортеж values.

						ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			28

### 3.2. Вирішення проблем з модулем

В розділі 3.1 була розібрана побудова програмного модулю, але були знайдені проблеми з можливим блокуванням доступу до сайту через надмірні запити до веб-сторінки тому далі розглянемо способи уникнення проблеми або дій у разі блокуванні

Для того аби на сайт приходили запити від імені користувача і не було автоматичного блокування ботів, замінемо headers.

Перейдемо у вікно інструментів розробників натисканням на клавішу f12 та перейдемо до вкладки Network як зазначено на рисунку

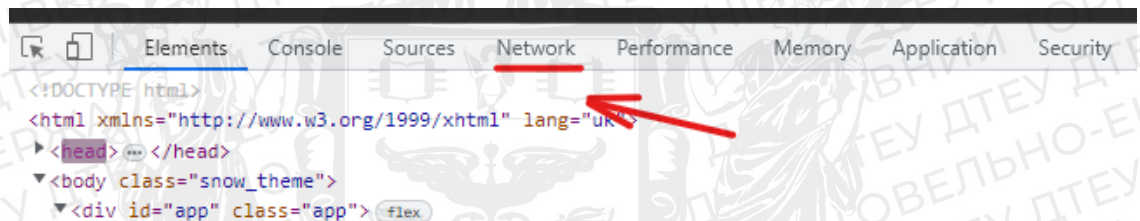
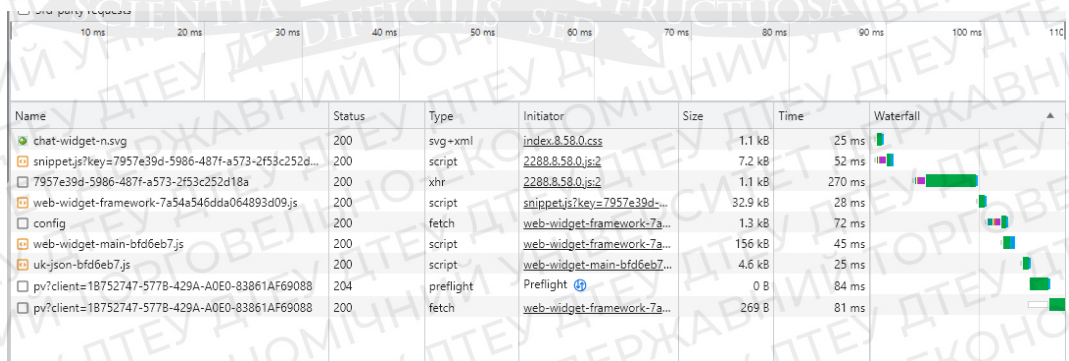


Рис. 3.6 вкладка Network

Після чого переходимо та оберемо будь-який get запит і знаходимо свій User Agent, який копіюємо.



Name	Status	Type	Initiator	Size	Time	Waterfall
chat-widget-n.svg	200	svg+xml	index.8.58.0.css	1.1 kB	25 ms	
snippets.js?key=7957e39d-5986-487f-a573-2f53c252d...	200	script	2288.8.58.0.js2	7.2 kB	52 ms	
7957e39d-5986-487f-a573-2f53c252d18a	200	xhr	2288.8.58.0.js2	1.1 kB	270 ms	
web-widget-framework-7a54a546dda064893d09.js	200	script	snippets.js?key=7957e39d-...	32.9 kB	28 ms	
config	200	fetch	web-widget-framework-7a...	1.3 kB	72 ms	
web-widget-main-bfd6eb7.js	200	script	web-widget-framework-7a...	156 kB	45 ms	
uk-json-bfd6eb7.js	200	script	web-widget-main-bfd6eb7...	4.6 kB	25 ms	
pv?client=18752747-5778-429A-A0E0-83861AF69088	200	preflight	Preflight	0 B	84 ms	
pv?client=18752747-5778-429A-A0E0-83861AF69088	200	fetch	web-widget-framework-7a...	269 B	81 ms	

Рис. 3.7 вкладка Network із запитами

						Аркуш
					ДТЕУ 121 06-26.БР	29
Зм.	Аркуш	№ докум	Підпис	Дата		



```

Referer: https://silpo.ua/
Sec-Ch-Ua: "Google Chrome";v="113", "Chromium";v="113", "Not-A.Brand";v="24"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-site
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
           Chrome/113.0.0.0 Safari/537.36

```

Рис. 3.8 особистий User-Agent в get запиті

Вийшов ось такий User-Agent:

```
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/113.0.0.0 Safari/537.36
```

Або можна скористатися User-Agent бібліотекою для генерації випадкових «агетів».

```

1 import requests
2 from requests.exceptions import RequestException
3
4 # посилання на сайт для парсингу
5 url = 'https://silpo.ua'
6
7 # додали headers, щоб сховати запит
8 headers = {
9     'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
10                  Chrome/113.0.0.0 Safari/537.36'
11 }
12
13 try:
14     # Надсилаємо GET-запит з вказаними заголовками
15     response = requests.get(url, headers=headers)
16
17     if response.status_code == 200:
18         content = response.content
19
20
21
22 except RequestException as e:
23     # Обробка помилки з'єднання або запиту
24     print("Помилка під час виконання запиту:", str(e))
25
26

```

Рис 3.9 код з заміною headers

Частина програмного коду, яка містить замінений headers на свій власний для запобігання блокування.

						Аркуш
						30
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 06-26.БР	

```

11
12 req = requests.get(url, headers=headers)
13 src = req.text
14 # print(src)
15
16 with open("index.html", "w") as file:
17     file.write(src)
18

```

Рис. 3.10 код для створення локальної версії сайту

Цей варіант підходить для ситуації коли на сайті отримали тимчасове блокування, але можна буде продовжити роботи зі збереженою сторінкою, яку можна також відкрити у браузері, вона збереже всі категорії та посилання.

### 3.3. Висновок до розділу 3

Отже, підсумовуючи вище викладене, в цьому розділі було розроблено програмний модуль парсингу. Були виявлені можливі проблеми з доступом до веб-сайту з якого виконується процес парсингу та розроблені рішення для запобігання виникненню подібного у майбутньому. Процес розробки був детально описаний з прикріпленням рисунків.

						ДТЕУ 121 06-26.БР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			31



## ВИСНОВКИ ТА ПРОПОЗИЦІЇ

З проведених досліджень можна зробити наступні висновки:

1. Проведено аналіз та огляд існуючих рішень;
2. В якості мови програмування було обрано Python, а в якості бази даних було обрано MySQL;
3. Були проведені моделювання з наведенням діаграм;
4. В процесі виконання проекту розроблено програмний модуль парсингу;
5. Розроблена система повністю задовольняє всім критеріям.



					<i>ДТЕУ 121 06-26.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпи</i>	<i>Дата</i>	Програмний модуль парсингу сайтів мереж продуктових магазинів	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Зав. каф.	Криворучко О.В.			28.04.23		<i>ВП</i>	32	50
Керівник	Власенко Л.О.			28.04.23		Факультет інформаційних технологій 4 курс, 6 група		
Гарант	Рзаєва С.Л.			28.04.23				
Розробив	Шарапановський Д.О.			28.04.23	<i>Висновки та пропозиції</i>			

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Парсинг цін із сайту [Електронний ресурс]. - Режим доступу: <https://www.pricecontrol.com.ua/ua/vse-pro-shho-slid-znati-pri-parsingu-sajtiv/>
2. Найбільш вживані мови програмування [Електронний ресурс]. - Режим доступу: <https://www.statista.com/statistics/793628/worldwide-developer-survey-most-used-languages/>
3. Ранги баз даних [Електронний ресурс]. - Режим доступу: [https://db-engines.com/en/ranking\\_trend](https://db-engines.com/en/ranking_trend)
4. Парсинг сайтів [Електронний ресурс]. - Режим доступу: <https://web-promo.ua/ua/blog/parsing-sajtov-hto-eto-i-zachem-nuzhen/#perevagy-parsyngy>
5. Parsing Techniques [Електронний ресурс]. - Режим доступу: [https://doc.lagout.org/science/0\\_Computer%20Science/4\\_Theory%20of%20Computation/Compiler%20Design/Parsing%20Techniques%20-%20A%20Practical%20Guide%2C%202nd%20Edition.pdf](https://doc.lagout.org/science/0_Computer%20Science/4_Theory%20of%20Computation/Compiler%20Design/Parsing%20Techniques%20-%20A%20Practical%20Guide%2C%202nd%20Edition.pdf)
6. Beautiful Soup Documentation [Електронний ресурс]. - Режим доступу: <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>
7. База даних MySQL [Електронний ресурс]. - Режим доступу: <https://promoter.net.ua/articles/baza-danix-mysql.html>

<i>ДТЕУ 121 06-26.БР</i>				
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпи</i>	<i>Дата</i>
Зав. каф.		Криворучко О.В.		23.12.22
Керівник		Власенко Л.О.		23.12.22
Гарант		Рзаєва С.Л.		23.12.22
Розробив		Шарапановський Д.О.		23.12.22
Програмний модуль парсингу сайтів мереж продуктових магазинів				
<i>Список використаних джерел</i>				
		<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
		<i>СВД</i>	<i>33</i>	<i>50</i>
Факультет інформаційних технологій 4 курс, 6 група				



## ДОДАТКИ

### ДОДАТОК А

#### Частина коду Requests

```
import requests

# посилання на сайт для парсингу
url = "https://silpo.ua"

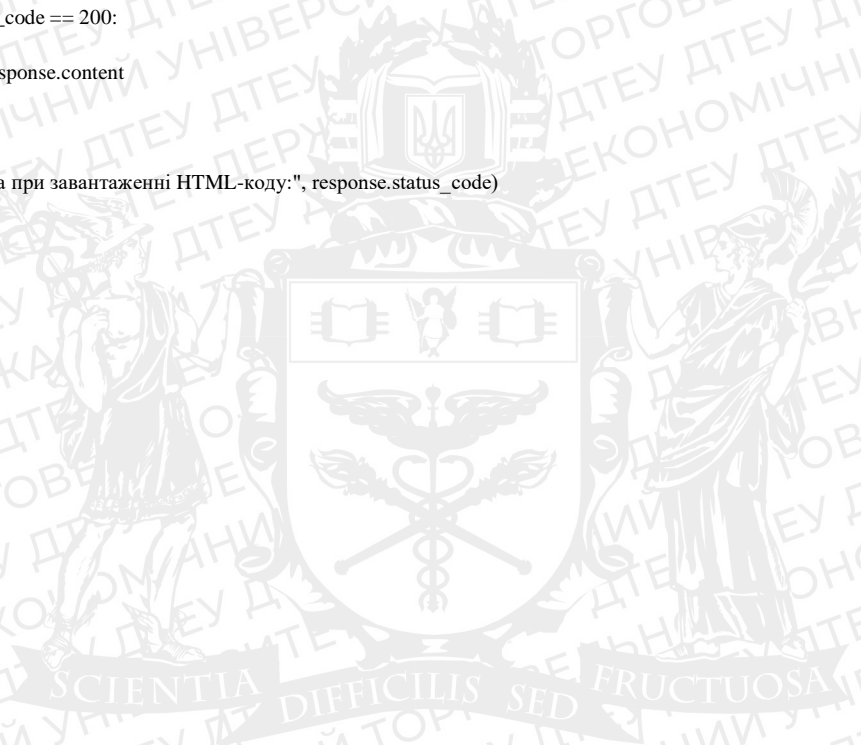
response = requests.get(url)

if response.status_code == 200:

    html_code = response.content

else:

    print("Помилка при завантаженні HTML-коду:", response.status_code)
```



## Requests з обходом блокування

```
import requests
from requests.exceptions import RequestException
import time

# посилання на сайт для парсингу
url = 'https://silpo.ua'
# додали headers, щоб сховати запит
headers = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/113.0.0.0 Safari/537.36'
}

try:
    time.sleep(5) # Затримка на 5 секунд
    response = requests.get(url, headers=headers)
    if response.status_code == 200:
        content = response.content
    except RequestException as e:
        # Обробка помилки з'єднання або запиту
        print("Помилка під час виконання запиту:", str(e))
```



**Код для парсинга назви та ціни**

```
from bs4 import BeautifulSoup
```

```
soup = BeautifulSoup(html_code, "html.parser")
```

```
h1_tags = soup.find_all("h1")
```

```
# Виведемо назву товару і його ціну
```

```
for h1_tag in h1_tags:
```

```
    title = h1_tag.text.strip() # Отримуємо текст заголовка, видаляємо зайві пробіли
```

```
    price_tag = h1_tag.find_next_sibling("span", class_="price")
```

```
    # Знаходимо наступний сусідній елемент <span> з класом "price"
```

```
    if price_tag:
```

```
        price = price_tag.text.strip() # Отримуємо текст ціни і видаляємо зайві пробіли
```

```
        print("Назва товару:", title)
```

```
        print("Ціна:", price)
```

```
        print("-----")
```

SCIENTIA DIFFICILIS SED FRUCTUOSA

**Код для збереження інформації до MySQL**

```
import mysql.connector

def insert_data(product_name, price, description):
    cursor = connection.cursor()
    sql = "INSERT INTO products (product_name, price, description) VALUES (%s, %s, %s)"
    values = (product_name, price, description)
    cursor.execute(sql, values)
    connection.commit()
    cursor.close()

insert_data(product_name, price, description)
```





**Частина коду для збереження локальної копії сторінки**

```
req = requests.get(url, headers=headers)
src = req.text
# print(src)
with open("index.html", "w") as file:
    file.write(src)
```

