

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавр

(ступінь вищої освіти)

на тему ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ЗБИРАННЯ
ДАНИХ З ВЕБСАЙТУ
SOFTWARE TO COLLECT DATA FROM A WEBSITE

Виконав(ла): студент(ка) 4 курсу, групи КНТ-229

Спеціальності 122 Комп'ютерні науки

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Комп'ютерні науки

КОЛБАСА Д.О.

(ПРИЗВИЩЕ та ініціали)

Керівник ГОФМАН Є.О.

(ПРИЗВИЩЕ та ініціали)

Рецензент СКРУПСЬКИЙ С.Ю.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет КНТ
Кафедра програмних засобів
Ступінь вищої освіти бакалавр
Спеціальність 122 Комп'ютерні науки
(код і найменування)

Освітня програма (спеціалізація) Комп'ютерні науки
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
Сергій СУББОТІН
“ ” 2023 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

КОЛБАСИ Дмитра Олександровича

(ПРІЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Програмне забезпечення для збирання даних з вебсайту. Software to Collect Data from a Website

керівник проєкту (роботи) к.т.н., ГОФМАН Євгеній Олександрович,
(науковий ступінь, вчене звання, ПРІЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від “ 04 ” квітня 2023 року № 87

2. Строк подання студентом проєкту (роботи) 01 червня 2023 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Матеріали і методи. 3. Опис програми. 4. Експлуатація, тестування та експериментальне дослідження програми.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)

Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	ГОФМАН Є.О., доцент		
Нормоконтроль	ЛИПОВЕЦЬ М.В., асистент		

7. Дата видачі завдання “4” квітня 2023 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	1 тиждень	Розділ 1
3	Вибір мови програмування та інших технологій розробки.	2 тиждень	Розділ 2
4	Розробка структури програми.	2 тиждень	Розділ 3
5	Розробка програми.	3-4 тижні	Розділи 3, 4
6	Тестування та експериментальне дослідження програмного забезпечення.	5 тиждень	Розділ 4
7	Оформлення пояснювальної записки та документів до неї.	6 тиждень	Додатки
8	Нормоконтроль та рецензування.	7 тиждень	
9	Захист роботи.	8 тиждень	

Студент(ка)

_____ Дмитро КОЛБАСА
(підпис) (Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)

_____ Євгеній ГОФМАН
(підпис) (Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра:
80 с., 3 табл., 30 рис., 3 дод., 21 джерело.

PYTHON, WEB-SCRAPPER, БАЗА ДАНИХ, ПАРСЕР, ПРОГРАМА-ЗБИРАЧ.

Об'єкт дослідження – процес збирання даних.

Предмет дослідження – програмні засоби для збирання даних.

Метою роботи є розробка програмного забезпечення для збирання даних з вебсайту.

Матеріали, методи та технічні засоби: мова програмування Python, середовище розробки Visual Studio Code, фреймворк Scrapy, система керування базами даних MySQL.

Результати. Розроблено програмне забезпечення для збирання даних з вебсайту за допомогою мови програмування Python, що дозволяє забезпечити збирання даних, таких як номери телефонів, електронні пошти, назви та адреси підприємств з сайту.

Практична цінність роботи полягає у розробці програмного забезпечення для збирання даних з вебсайту.

Висновки. Виконано проєктування програмного забезпечення для збирання даних з вебсайту. Розроблено програмне забезпечення для вебпарсингу. Здійснено тестування розробленого програмного забезпечення для збирання даних з вебсайту.

Галузь використання – аналіз даних вебсайтів.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 80 pages, 3 tables, 30 figures, 3 appendixes, 21 sources.

PYTHON, WEB-SCRAPER, DATABASE, PARSER, ASSEMBLER.

The object of research is the process of data collection.

The subject of the research is software for data collection.

The purpose of this work is to develop the software for data collection from the website.

Materials, methods and technical tools: Python programming language, Visual Studio Code development environment, Scrapy framework, MySQL database management system.

Results. The software tools for i website data collection based on the Python programming language, which allows the collection of data such as phone numbers, e-mails, business names and addresses from the website have been developed.

The practical value of the work lies in the development of software for data collection.

Conclusions. The software for data collection has been designed. The software for web parsing has been developed. The developed software for data collection has been tested.

Scope of use – website data analysis.

ЗМІСТ

	С.
Перелік скорочень та умовних познач	8
Вступ.....	9
1 Аналіз предметної області.....	11
1.1 Збирання даних з вебсайтів	11
1.2 Дослідження програмних засобів для збирання даних з вебсайтів.....	19
1.3 Висновки за розділом 1	29
2 Матеріали і методи	31
2.1 Вибір мови програмування та інших технологій розробки	31
2.2 Вибір середовища розробки для створення програмного забезпечення збирання даних з вебсайту	35
2.3 Висновки за розділом 2	37
3 Опис програми	39
3.1 Структура процесів програми збирання даних з вебсайту.....	39
3.2 Опис програмних модулів процесу генерації посилань	40
3.3 Опис програмних модулів процесу парсингу.....	40
3.4 Архітектура програмного забезпечення для збирання даних з вебсайту.....	41
3.5 Структура бази даних	43
3.6 Файлова структура проєкту програмного забезпечення для збирання даних з вебсайту.....	44
3.7 Висновки за розділом 3	48
4 Експлуатація, тестування та експериментальне дослідження програми.....	50
4.1 Призначення й умови застосування програми	50
4.2 Характеристики програми та її встановлення	51
4.3 Інструкція по експлуатації програми.....	51
4.3.1 Звернення до програми.....	51
4.3.2 Вхідні й вихідні дані	52
4.3.3 Повідомлення.....	52
4.4 Виконання програмного забезпечення для збирання даних з вебсайту ..	52

4.5 Висновки за розділом 4	55
Висновки	57
Перелік джерел посилань	59
Додаток А Технічне завдання	61
Додаток Б Фрагмент тексту програми	65
Додаток В Слайди презентації	73

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

API – програмний інтерфейс додатку;

IDE – інтегрована середовище розробки;

БД – база даних;

ПЗ – програмне забезпечення;

ПК – персональний комп'ютер;

СКБД – систем управління базою даних;

ТЗ – технічне завдання.

ВСТУП

У наш час широко застосовуються засоби вебскрапінгу, тобто збирання даних з вебсайтів, що дозволяє видобувати та систематизувати велику кількість вебданих, виконувати вебіндексування, аналіз даних, моніторинг змін цін і порівняння цін, збирання оглядів продуктів, виконувати збір списків нерухомості, моніторинг погодних даних, виявлення змін вебсайтів, відстеження присутності та репутації в Інтернеті, а також розв'язувати інші завдання.

Існує багато програм для збирання даних з вебсайтів. Проте програми збору даних можуть порушувати авторські права та викликати проблеми з безпекою даних. Крім того, універсальні програми збирання даних можуть мати проблеми, пов'язані зі заборонаю або обмеженням доступу, оскільки деякі вебсайти можуть мати правила, які обмежують доступ або забороняють програмам збирати їх дані. Це може призвести до блокування програми або зниження її продуктивності. Також існують проблеми з обмеженням на кількість запитів, низькою якістю даних, а також зі зміною структури вебсайту, внаслідок чого програма збирання даних може не зможти знайти та зібрати необхідні дані.

Тому актуальною є розробка програмного забезпечення для збирання даних з вебсайту

У дипломній кваліфікаційній роботі бакалавра розв'язується актуальне завдання розробки програмного забезпечення для збирання даних з вебсайту.

Для досягнення поставленої мети у кваліфікаційній роботі бакалавра необхідно розв'язати такі задачі:

- виконати аналіз предметної області та програмних засобів для збирання даних з вебсайту;
- здійснити проєктування програмного забезпечення для збирання даних з вебсайту;
- створити програмне забезпечення для збирання даних з вебсайту;

– виконати тестування розробленого програмного забезпечення для збирання даних з вебсайту.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Збирання даних з вебсайтів

Деякі вебсторінки можуть містити багато корисної інформації [1]-[5], зокрема, курси акцій, деталі, спортивна статистика, корпоративні комунікації тощо. Щоб отримати доступ до цієї інформації, необхідно використовувати формат, який використовує сайт, або вручну скопіювати та вставити інформацію в новий документ [1]-[5].

Для вирішення таких задач використовуються методи збирання даних з вебсайтів (вебскрапінг, web scraping). Ця інформація збирається та надсилається користувачам у корисній формі, наприклад, у вигляді таблиці або за допомогою API [1]-[5].

Хоча вебзбирання можна виконувати вручну, у більшості випадків для вебзбирання надають перевагу автоматизованим інструментам, оскільки вони дешевші та можуть працювати швидше [1]-[5].

Однак у більшості випадків видобути інформацію з вебсайту досить непросто. Оскільки вебсайти мають різні форми та рішення, веббраузери відрізняються за функціями та функціями. Деякі сторінки можуть бути заблоковані, коли ви намагаєтеся їх проаналізувати, тому перед видобуванням інформації зі сторінки рекомендується виконати певні дії щодо того, як уникнути та обійти капчу [1]-[5].

Програми видобування даних спочатку отримують одну або кілька URL-адрес перед тим, потім програма зчитує весь HTML-код на цій сторінці. Найбільш просунуті розробники нададуть весь сайт, включаючи CSS і Javascript. Потім програма вебскрапінгу витягує всі дані зі сторінки або певні дані, вибрані користувачем перед запуском проєкту. В ідеалі користувач пройде процес вибору потрібної інформації зі сторінки. Наприклад, можна шукати ціни та моделі на сторінках продуктів Amazon, але не потрібно переглядати огляди продуктів. Далі, програмне забезпечення збирання даних представляє всі зібрані дані у зручний для користувача спосіб [1]-[5].

У той час як більшість веббраузерів виводять дані у форматі CSV або таблиці Excel, багато вдосконалених браузерів підтримують інші формати, доступні через API, наприклад за допомогою формату JSON [1]-[5].

Для простоти програми видобування даних з сайту можна класифікувати за чотирма критеріями або категоріями [1]-[5]:

- програми, розроблені самостійно або заздалегідь;
- програми у вигляді розширень браузера чи повноцінне програмне забезпечення;
- тип користувацького інтерфейсу;
- хмарні та локальні.

Так само, як можна створити власний вебсайт для власних потреб, так і можна створити програму-збирач даних для своїх задач. Однак інструменти, доступні для створення власного сайту копіювання, все одно вимагають розширених навичок програмування. Цей обсяг інформації збільшується разом із кількістю функцій, які необхідно мати у своєму вебскрабері. З іншого боку, є багато готових програм-збирачів даних, які можна завантажити та запустити одразу. Деякі з них включають розширені опції, такі як видобування даних за розкладом, експорт у JSON формат або формат Google таблиць тощо [1]-[5].

Загалом програми для збирання даних з вебсайтів бувають двох форм: розширення для браузерів або програмне забезпечення для настільних ПК. Розширення веббраузера – це схожі програми, які можна додати до свого веббраузера, наприклад Google Chrome або Firefox. До популярних розширень веббраузерів належать теми, блокування реклами, розширення для повідомлень тощо [1]-[5].

Програми для збирання даних з вебсайтів у вигляді розширень веббраузерів є простими у використанні та мають перевагу легкої інтеграції у браузер. Однак ці розширення зазвичай обмежуються терміном служби браузера. Це означає, що неможливо реалізувати всі розширені функції, які потрібно виконувати поза браузером. Наприклад, таким розширенням не можна призначити IP-адресу [1]-[5].

З іншого боку, існує окреме програмне забезпечення для збирання даних з вебсайтів, яке можна завантажити та встановити на комп'ютер. Таке програмне забезпечення є меншим корисним, ніж розширення вебпереглядача, але вони роблять це сучасним способом, який не обмежується тим, що браузер може, а що ні [1]-[5].

Інтерфейс користувача може сильно відрізнятися в різних програмах для збирання даних з вебсайтів. Наприклад, деякі інструменти вебзбирання використовують невеликий інтерфейс користувача та вебінтерфейс. Деякі користувачі можуть вважати це недоречним або заплутаним. З іншого боку, деякі програми для збирання даних з вебсайтів мають повний інтерфейс користувача, який відображає всю сторінку, щоб користувач міг обрати інформацію, яку він хоче проаналізувати та дослідити. Ці програми для збирання даних з вебсайтів прості у використанні для більшості людей з невеликими технічними знаннями [1]-[5].

Деякі програми для збирання даних з вебсайтів навіть заходять так далеко, що консультують і допомагають користувачам через інтерфейс користувача зрозуміти все, що може запропонувати програмне забезпечення [1]-[5].

Просту програму для збирання даних з вебсайтів можна запустити на комп'ютері за допомогою обладнання та підключення до Інтернету. Це означає, що якщо програма для збирання даних з вебсайтів працює, то вона буде споживати багато ресурсів комп'ютера (процесор та оперативна пам'ять), внаслідок чого комп'ютер може сповільнюватися під час роботи. Крім того, якщо програма для збирання даних з вебсайтів налаштована на роботу з великою кількістю URL-адрес (наприклад, сторінок продуктів), це може вплинути на пропускну здатність та можливості провайдера [1]-[5].

Хмарні програми для збирання даних з вебсайтів зазвичай працюють на сторонніх серверах, наданих компанією, яка їх розробила. Це означає, що ресурси персонального комп'ютера звільняються, поки така програма працює та збирає дані. Тому користувач може працювати над іншими завданнями та

отримувати сповіщення, коли нотатки будуть готові до експорту. Це дозволяє легко інтегрувати розширені функції, такі як ротація IP-адрес, що запобігає блокуванню ресурсів програмою для збирання даних з великих вебсайтів через активність та складність цього процесу [1]-[5].

У наш час програми для збирання даних з вебсайтів набули широкого практичного використання у таких сферах [1]-[5]:

- аналіз інформації про нерухомість. Багато агентів з нерухомості використовують вебсайти оголошень для розміщення оголошень про продаж чи оренду нерухомості. Наприклад, ріелтор за допомогою відповідної програми видобуває інформацію зі списку нерухомості, виставленої на продаж, щоб створити API для публікації цієї інформації на своєму вебсайті. Багато списків, які користувач може знайти на вебсайтах нерухомості, можуть бути автоматично створені через API відповідних програм [1]-[5];

- статистика та галузева статистика. Багато компаній використовують програми для збирання даних з вебсайтів для створення великих баз даних і отримання інформації з них. Ці компанії можуть продавати отриману та проаналізовану інформацію. Наприклад, компанія може проаналізувати велику кількість даних про ціни, експорт та імпорт нафти, щоб надати інформацію нафтовим компаніям у всьому світі [1]-[5];

- порівняння торгових сайтів. Деякі вебсайти та програми допомагають легко порівнювати ціни кількох продавців одного продукту. Одним із способів роботи цих сайтів є використання скребка для щоденного збирання інформації про продукт і цін від кожного продавця. Таким чином вони можуть надати своїм клієнтам корисну порівняльну інформацію [1]-[5];

- генерація потенційних клієнтів. Програми для збирання даних з вебсайтів використовується багатьма компаніями для збору інформації про клієнтів або потенційних клієнтів [1]-[5].

Отже, вебзбирання — це процес автоматичного збору інформації з вебсайту. Його також називають видобуванням даних з веб (web data extraction, web data scraping) [1]-[5].

Вебзбирання зазвичай використовується окремими особами та компаніями, які хочуть використовувати загальнодоступні онлайн дані, щоб надавати інформацію та приймати розумніші рішення [1]-[5].

На відміну від складного процесу ручного вилучення даних, вебзбирання використовує інтелектуальну автоматизацію для відновлення сотень, мільйонів або навіть мільярдів даних із, здавалося б, величезних частин Інтернету [1]-[5].

Програма для збирання даних (вебскребок, web scraper) запускає процес швидкого та ефективного вилучення інформації з інших вебсайтів. Отримані дані представлені в структурованому форматі, що полегшує аналіз і використання у ваших проєктах [1]-[5].

Вебсканер (web crawler), також відомий як «павук», є системою на основі штучного інтелекту, що здійснює пошук в Інтернеті, щоб відобразити та знайти вміст, переходячи за посиланнями та здійснюючи пошук. Вебскребок — це спеціальний інструмент, призначений для ефективного та швидкого отримання інформації з вебсайтів. Залежно від проєкту інструменти вебзбирання значно відрізняються за дизайном і складністю [1]-[5].

Важливою частиною будь-якого вебскребка є джерело даних (або селектор), яке він використовує для пошуку інформації, що хоче отримати з файлу HTML, часто за допомогою XPath, селекторів CSS, регулярного виразу або їх комбінації [1]-[5].

Узагальнено процес видобування даних з вебсайтів може бути поданий такими етапами [1]-[5]:

- визначення цільового вебсайту;
- збір URL-адрес сторінок, з яких необхідно видобути дані;
- запит за URL-адресами для одержання HTML-коду відповідних сторінок;
- використання локаторів (locators) для пошуку інформації в HTML;
- збереження даних у зручних форматах, наприклад, у виглядіJSON або CSV файлах або в інших структурованих форматах [1]-[5].

Є багато інструментів для видобування даних з відкритим кодом, якими можна скористатися, але всі вони мають свої обмеження [1]-[5].

Інструменти для вебзбирання являють собою програмне забезпечення, призначене для вилучення даних із вебсторінок. Коли ви збираєте дані з вебсайту, ви майже напевно використовуєте певну форму вебзбирання для отримання особистої інформації [1]-[5].

Деякі інструменти використовуються як частина програми вебзбирання для сканування запитів HTTP від цільового вебсайту та вилучення даних вебсайту зі сторінки. Він визначає вміст, який є загальнодоступним, видимим для користувачів і відображається сервером як HTML [1]-[5].

У деяких випадках API вимагає, щоб відповідна інформація, як-от ціна продукту та деталі, зберігалася в базі даних і передавалася в браузер за допомогою запиту HTTP. Існують різні види інструментів видобування даних з вебсайтів, які можна налаштувати для різних проєктів вилучення даних [1]-[6].

Інструменти збирання даних з вебсайту можуть бути важливим набором інструментів для будь-якого простого завдання видобування та аналізу даних, крім того, також можна використовувати бібліотеки загального призначення та зв'язати їх разом, щоб створити програми вебзбирання даних [1]-[6].

Наприклад, можна використовувати бібліотеку запитів HTTP (наприклад, бібліотеку запитів Python), щоб отримати дані з необхідної сторінки та інтегрувати їх із бібліотекою BeautifulSoup Python. Крім того, можна використовувати спеціальну структуру, яка інтегрує клієнт HTTP з бібліотекою HTML [1]-[6].

Для всіх проєктів, крім програмних проєктів малого бізнесу, як правило, необхідними є засоби збирання даних з вебсайту або програмне забезпечення для аналізу даних, щоб отримати дані з Інтернету [1]-[6].

Засоби збирання даних з вебсайту надають структуровані вебдані з будь-якого загальнодоступного вебсайту. Окрім сучасних зручностей,

справжня сила хмарних обчислень для збирання вебданих полягає в їхній здатності створювати та підтримувати найбільш революційні бізнес-процеси у світі. Багато сучасних компаній використовують дані, зібрані з Інтернету, щоб покращити свою діяльність, інформуючи керівництво про рішення аж до індивідуального досвіду обслуговування клієнтів [1]-[6].

Засоби збирання даних з вебсайту дозволяють отримувати інформацію про різні продукти та ціни з сайтів електронної комерції, розумне перетворення відповідної інформації є важливою частиною сучасних компаній електронної комерції, які прагнуть приймати кращі рішення на основі даних і ціноутворення на рекламу. Отримані за допомогою засобів збирання даних з вебсайту дані про ціни в Інтернеті надають такі переваги: можливість встановлення хорошої ціни, що призведе до поліпшення доходу, моніторинг конкурентів, можливість моніторингу тенденцій на ринку відносно конкретного продукту, забезпечення відповідності бренду [1]-[6].

Засоби збирання даних з вебсайтів дозволяють забезпечити дослідження ринків, що є важливим та має базуватися на якнайточнішій інформації. Збираючи дані, користувач отримує глибоке розуміння якості, кількості та глибини вебсайтів усіх форм і розмірів, створюючи аналіз глобального ринку та бізнес-аналітику. Впроваджуючи отримані з Інтернету дані про продукти в повсякденний бізнес, агенти та брокерські компанії можуть захистити себе від низхідної онлайн-конкуренції та приймати обґрунтовані рішення на ринку [1]-[6].

Засоби збирання даних з вебсайтів дозволяють виконувати моніторинг новин і контенту. Сучасні засоби масової інформації можуть створити величезну цінність для бізнесу або створити потенційну загрозу за допомогою одного циклу новин. Тому компанії, які залежить від своєчасного аналізу новин, та ті, які часто з'являється в новинах, потребують аналізу даних у вигляді новин, що є найкращим рішенням для моніторингу, агрегування та аналізу найважливіших історій у відповідній галузі [1]-[6].

Такі засоби часто використовуються для автоматизації бізнесу у випадках, коли необхідно отримати доступ до даних у структурованому вигляді, що можна зробити з даних вебсайту [1]-[6].

У наш час існують різні рішення для збирання вебданих, доступних для автоматизації процесу збирання вмісту та видобування даних з вебсайтів. Вони варіюються від простих рішень для копіювання за принципом «вклади та клацни», призначених для неспеціалістів, до потужніших програм, орієнтованих на розробників, із широкими можливостями налаштування та керування [1]-[6]. Програмне забезпечення для збирання даних з вебсайту дозволяє витягувати вебдані з кількох вебсторінок (іноді, тисяч або мільйонів сторінок на день) і організовувати результати в структурованому форматі. Існують також прості у користуванні рішення для збирання вебданих, які працюють як розширення браузера або плагіни із простим інтерфейсом «вклади та клацни». Менш складні, ніж їхні аналоги, зосереджені на розробниках, вони зазвичай більш обмежені в різноманітності та обсязі даних, які вони дозволяють збирати [1]-[6].

Для видобування даних з вебсайтів можуть використовуватися різні програмні засоби, які відповідають інтересам і здібностям користувача [1]-[6].

Відзначимо, що у процесі розроблення програмного забезпечення для збирання даних з вебсайтів треба врахувати деякі особливості та фактори. Наприклад, під час написання вебскребка важливо якомога конкретніше визначити, що саме (які саме дані) необхідно збирати. Необхідно заздалегідь витратити трохи часу, щоб скласти чіткий план та визначити перелік даних, які необхідно видобути, що заощадить багато зусиль з очищення даних у довгостроковій перспективі [1]-[6].

Вебсайти можуть мати файл під назвою robots.txt, який призначено для взаємодії з програмами для збирання даних. У цьому файлі повідомляється інформація, які області вебсайту є обмеженими. Якщо згідно цього файлу забороняється копіювання інформації з певних сторінок вебсайту, то програмісту слід завжди дотримуватися цих інструкцій [1]-[6].

Окрім файлу robots.txt, необхідно також врахувати умови використання вебсайту. Умови надання послуг можуть містити правові положення, які визначають, що можна, а що не можна робити з інформацією на вебсайті. Якщо власник програми вебскрабінгу поруше ці правила, можна отримати проблеми з законом [1]-[6].

Також необхідно дотримуватися протоколів захисту даних, при чому дотримуватися протоколів захисту даних кожного регіону. Наприклад, у Європейському Союзі Загальний регламент захисту даних захищає певні персональні дані від збирання та аналізу, тобто незаконно збирати відповідні дані без явної згоди людей [1]-[6].

Визначено, що у наш час широко застосовуються засоби вебскрапінгу, тобто збирання даних з вебсайтів, що дозволяє видобувати та систематизувати велику кількість вебданих, виконувати вебіндексування, аналіз даних, моніторинг змін цін і порівняння цін, збирання оглядів продуктів, виконувати збір списків нерухомості, моніторинг погодних даних, виявлення змін вебсайтів, відстеження присутності та репутації в Інтернеті, а також розв'язувати інші завдання. Виконано класифікацію програм видобування даних з вебсайтів. Проаналізовано галузі використання програм для збирання даних з вебсайтів. Наведено етапи узагальненого процесу видобування даних з вебсайтів. Визначено фактори, які необхідно врахувати у процесі розроблення програмного забезпечення для збирання даних з вебсайтів.

1.2 Дослідження програмних засобів для збирання даних з вебсайтів

Зараз існує значна кількість програм для збирання даних з вебсайтів. Проведемо огляд певних програм та вебсайтів цієї галузі [7]-[16].

Програмне забезпечення Octoparse [7], [12] є програмою для збирання даних з вебсайтів. Програмне забезпечення Octoparse може використовуватися користувачами без навичок кодування. У процесі видобування даних з

вебсайтів при використанні програми Ostoparse користувачі мають контроль над усім процесом за допомогою простого у використанні інтерфейсу. Ostoparse є ефективним інструментом видобування даних з сайтів для користувачів, що не вміють програмувати, дозволяє користувачам видобувати дані з різних форм введення, заповнювати відповідні форми, вводити пошукові терміни, прокручувати дані, відтворювати JavaScript тощо. Це програмне забезпечення засновано на хмарних технологіях та містить аналізатор вебсайтів. Ostoparse (рис. 1.1) поставляється з досить продуктивним безкоштовним рівнем, який дозволяє користувачам створювати до 10 сканерів безкоштовно. Для корпоративних клієнтів пропонуються повністю налаштовані сканери та керовані рішення.

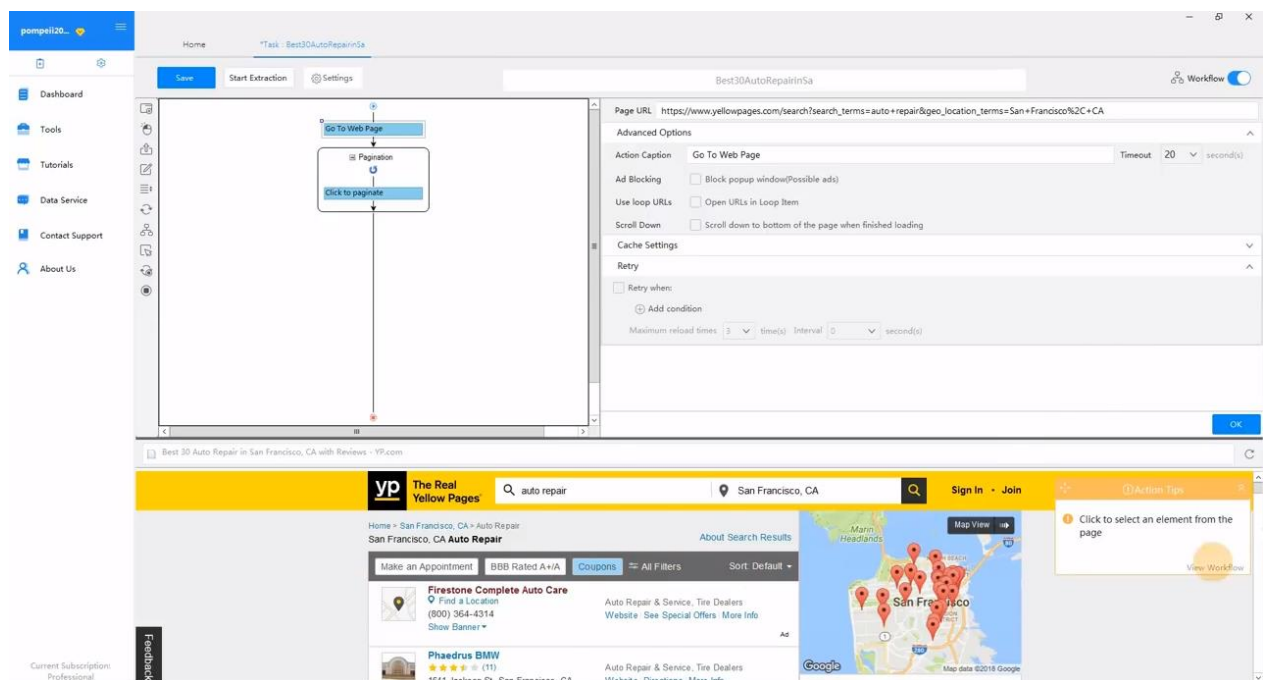


Рисунок 1.1 – Програмне забезпечення Ostoparse [7], [12]

Серед особливостей даного сервісу, що надає можливості для збирання даних з вебсайту, можна виділити такі [7], [12]:

- можливість збирання даних з вебсайтів через зручний інтерфейс;
- відсутність необхідності навичок програмування у користувачів;
- можливість вилучення довільного елемента HTML;

- видобування списків вебсторінок і вилучення відповідних детальних сторінок;
- можливість оброблення великих обсягів даних;
- можливість застосування регулярних виразів;
- засоби API;
- ефективна служба підтримки;
- засоби збереження даних у форматі CSV, XML, HTML тощо;
- наявність безкоштовного тарифного плану;
- використання хмарних технологій.

Серед недоліків програмного забезпечення Octoparse виділяється те, що ця програма може працювати тільки під операційною системою Windows або macOS, а також платне користування [7], [12]. Крім того, відсутня україномовна версія цього програмного забезпечення [7], [12].

Програмне забезпечення Import.io є безкоштовним програмним забезпеченням для вебскребування, яке дозволяє видобувати дані з вебсайтів і організовувати їх в набори даних. Це програмне забезпечення має сучасний інтерфейс, що полегшує використання [7], [13]. Програмне забезпечення Import.io [7], [13] є програмою для збирання даних з вебсайтів. може використовуватися користувачами без навичок кодування. У процесі видобування даних з вебсайтів при використанні програми Import.io користувачі мають контроль над усім процесом за допомогою простого у використанні інтерфейсу. Import.io є ефективним інструментом видобування даних з сайтів для користувачів, що не вміють програмувати, дозволяє користувачам видобувати дані з різни форм введення, заповнювати відповідні форми, вводити пошукові терміни, прокручувати дані, відтворювати JavaScript тощо. Це програмне забезпечення засновано на хмарних технологіях та містить аналізатор вебсайтів. Import.io (рис. 1.2) поставляється з досить продуктивним безкоштовним рівнем. Для корпоративних клієнтів пропонуються повністю налаштовані сканери та керовані рішення [7], [13].

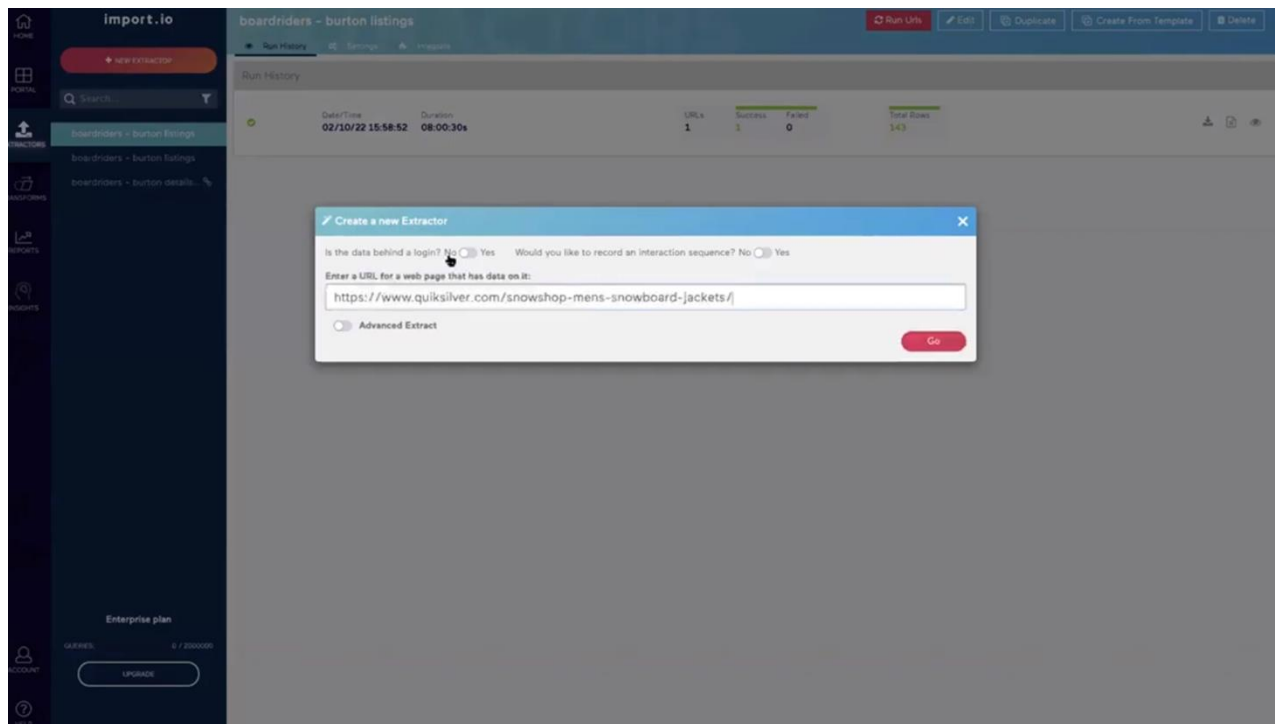


Рисунок 1.2 – Програмне забезпечення Import.io [7], [13]

Серед особливостей сервісу Import.io, що надає можливості для збирання даних з вебсайту, можна виділити такі [7], [13]:

- можливість збирання даних з вебсайтів через зручний інтерфейс;
- відсутність необхідності навичок програмування у користувачів;
- можливість вилучення довільного елемента HTML;
- видобування списків вебсторінок і вилучення відповідних детальних сторінок;
- можливість оброблення великих обсягів даних;
- можливість застосування регулярних виразів;
- ефективна служба підтримки;
- засоби збереження даних у форматі CSV, XML, HTML тощо;
- наявність безкоштовного тарифного плану;
- онлайн-сховище даних;
- використання хмарних технологій.

Серед недоліків програмного забезпечення Import.io виділяється те, що ця програма в деяких випадках може некоректно збирати дані, а також платне

користування [7], [13]. Крім того, відсутня україномовна версія цього програмного забезпечення [7], [13].

Програмне забезпечення Mozenda (рис. 1.3) забезпечує інструмент видобування даних, що дозволяє легко видобувати вміст вебсайтів з Інтернету. Це програмне забезпечення вебскребування точкових даних та кліків по певним об'єктам керування [7], [14].

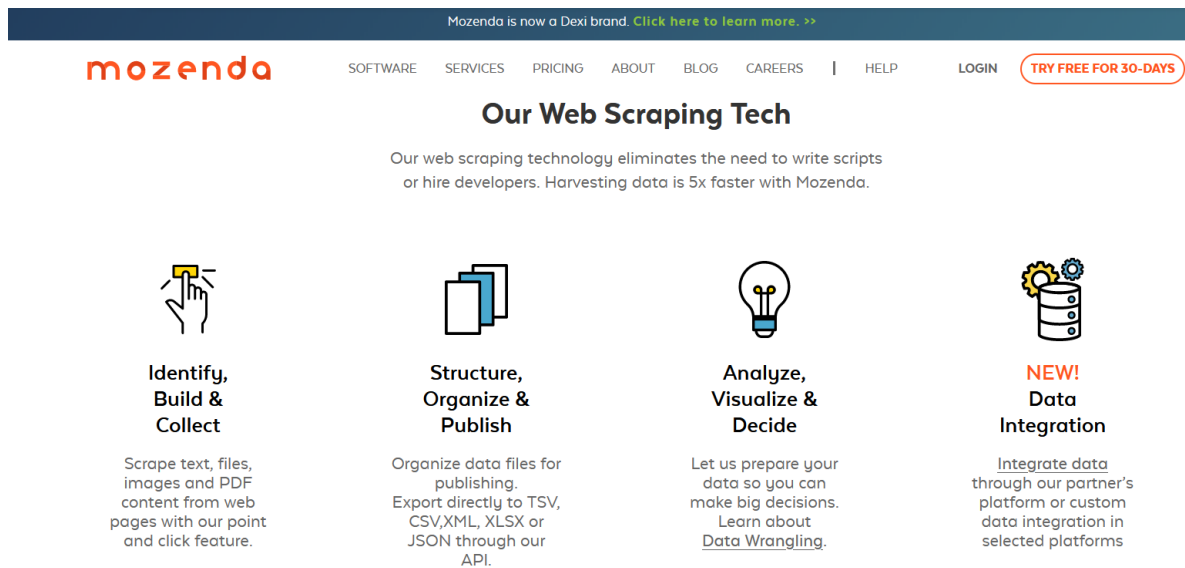


Рисунок 1.3 – Програмне забезпечення Mozenda [7], [14]

Програмне забезпечення Mozenda має сучасний інтерфейс, що полегшує використання [7], [14]. Програмне забезпечення Mozenda [7], [14] є програмою для збирання даних з вебсайтів може використовуватися користувачами без навичок кодування. У процесі видобування даних з вебсайтів при використанні програми Mozenda користувачі мають контроль над усім процесом за допомогою простого у використанні інтерфейсу. Mozenda є ефективним інструментом видобування даних з сайтів для користувачів, що не вміють програмувати, дозволяє користувачам видобувати дані з різни форм введення, заповнювати відповідні форми, вводити пошукові терміни, тощо. Це програмне забезпечення засновано на хмарних технологіях та містить аналізатор вебсайтів [7], [14].

Програмне забезпечення Mozenda являє собою хмарну самообслуговувану платформу копіювання вебсторінок, Mozenda має досвід обслуговування корпоративних клієнтів з усього світу. Mozenda вирізняється своєю службою обслуговування клієнтів (надає підтримку як телефоном, так і електронною поштою всім платним клієнтам). Платформа має високу масштабованість і також дозволяє розміщувати локальний хостинг. Найнижчий план починається від 250 доларів США на місяць [7], [14].

Серед особливостей сервісу Mozenda, що надає можливості для збирання даних з вебсайту, можна виділити такі [7], [14]:

- можливість збирання даних з вебсайтів через зручний інтерфейс;
- відсутність необхідності навичок програмування у користувачів;
- можливість вилучення довільного елемента HTML;
- видобування списків вебсторінок і вилучення відповідних детальних сторінок;
- можливість оброблення великих обсягів даних;
- можливість застосування регулярних виразів;
- засоби API;
- ефективна служба підтримки;
- засоби збереження даних у форматі CSV, XML, HTML тощо;
- онлайн-сховище даних;
- використання хмарних технологій.

Серед недоліків програмного забезпечення Mozenda виділяється те, що ця програма в деяких випадках може некоректно збирати дані, а також відсутність безкоштовних тарифних планів [7], [14]. Крім того, відсутня україномовна версія цього програмного забезпечення [7], [14].

Програмне забезпечення ParseHub [7], [15] є візуальним програмним забезпеченням, яке можна використовувати для отримання даних з Інтернету. Користувач може легко створювати API з вебсайтів, які не надають їх. ParseHub (рис. 1.4) є досить потужним інструментом для створення

вебскребків без програмування. Ним користуються аналітики, журналісти, спеціалісти з обробки даних [7], [15].

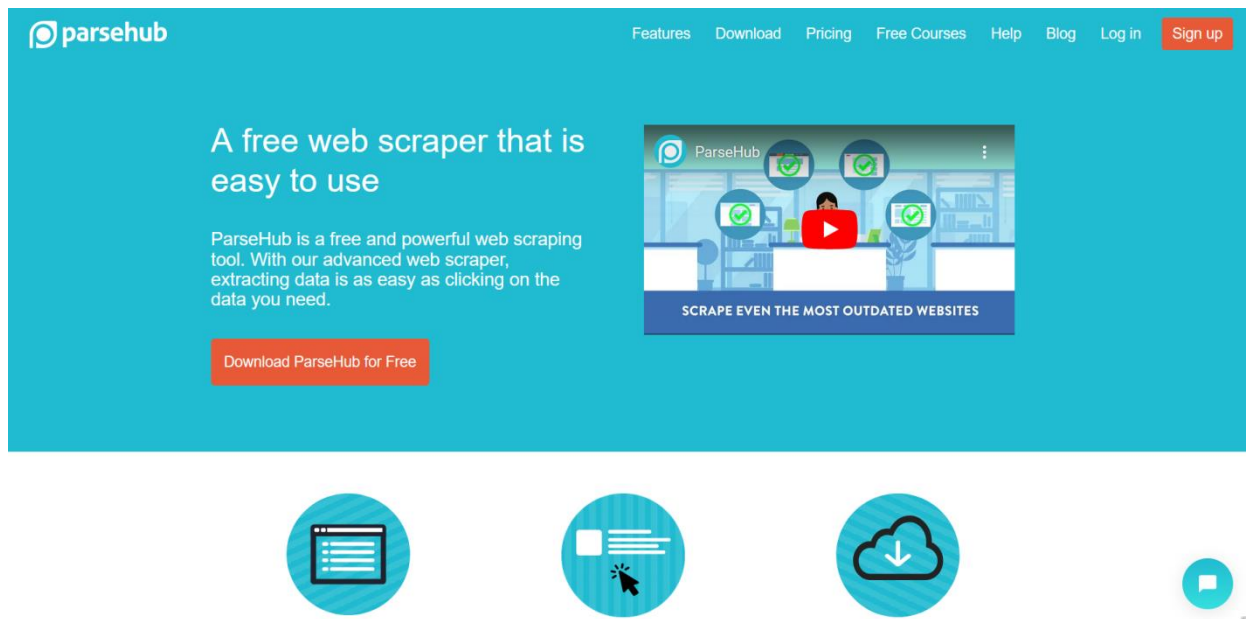


Рисунок 1.4 – Програмне забезпечення ParseHub [7], [15]

Програмне забезпечення ParseHub є досить простим у використанні. У користувача є можливість створювати вебскрепки, натиснувши на потрібні дані. Потім Parsehub експортує дані у формат JSON або Excel. У ньому є багато зручних функцій, таких як автоматична ротація IP-адреси, можливість перегляду вебсторінок за критеріями входу, перегляду спадних меню та вкладок, отримання даних із таблиць і карт та багато іншого. Крім того, він має безкоштовний тарифний, що дозволяє користувачам очищати до 200 сторінок даних лише за 40 хвилин. ParseHub також хороший тим, що він надає настільні клієнти для Windows, Mac OS і Linux, тому є можливість використовувати цю програму незалежно від операційної системи [7], [15].

Програмне забезпечення ParseHub має сучасний інтерфейс, може використовуватися користувачами без навичок кодування. У процесі видобування даних з вебсайтів при використанні програми ParseHub користувачі мають контроль над усім процесом за допомогою простого у

використанні інтерфейсу. ParseHub є ефективним інструментом видобування даних з сайтів для користувачів, що не вміють програмувати, дозволяє користувачам видобувати дані з різни форм введення, заповнювати відповідні форми, вводити пошукові терміни, тощо. Це програмне забезпечення засновано на хмарних технологіях та містить аналізатор вебсайтів [7], [15].

Серед особливостей сервісу ParseHub, що надає можливості для збирання даних з вебсайту, можна виділити такі [7], [15]:

- можливість збирання даних з вебсайтів через зручний інтерфейс;
- відсутність необхідності навичок програмування у користувачів;
- можливість вилучення довільного елемента HTML;
- видобування списків вебсторінок і вилучення відповідних детальних сторінок;
- можливість оброблення великих обсягів даних;
- можливість застосування регулярних виразів;
- засоби API;
- ефективна служба підтримки;
- засоби збереження даних у форматі CSV, XML, HTML тощо;
- онлайн-сховище даних;
- наявність безкоштовного тарифного плану;
- використання хмарних технологій.

Серед недоліків програмного забезпечення ParseHub виділяється те, що ця програма в деяких випадках може некоректно збирати дані [7], [15]. Крім того, відсутня україномовна версія цього програмного забезпечення [7], [15].

Програмне забезпечення Diffbot – це рішення корпоративного рівня для компаній, які потребують високоспеціалізованого сканування даних і сканування екрана, особливо для тих, хто сканує вебсайти, які часто змінюють структуру HTML [7], [16].

Програмне забезпечення Diffbot (рис. 1.5) відрізняється від більшості інструментів для сканування вебсторінок тим, що він використовує комп'ютерне бачення (замість синтаксичного аналізу HTML) для визначення

релевантної інформації на сторінці. Це означає, що навіть якщо структура HTML сторінки зміниться, вебскрепки користувача не зламаються, доки сторінка візуально виглядає так само. Це є критично важливою функцією для тривалих робіт із збирання даних з вебсторінок. Програмне забезпечення Diffbot є досить дорогим (най дешевший тарифний план — 299 доларів США на місяць), але він чудово працює, пропонуючи преміум-сервіс, який, можливо, коштує того для великих клієнтів [7], [16].

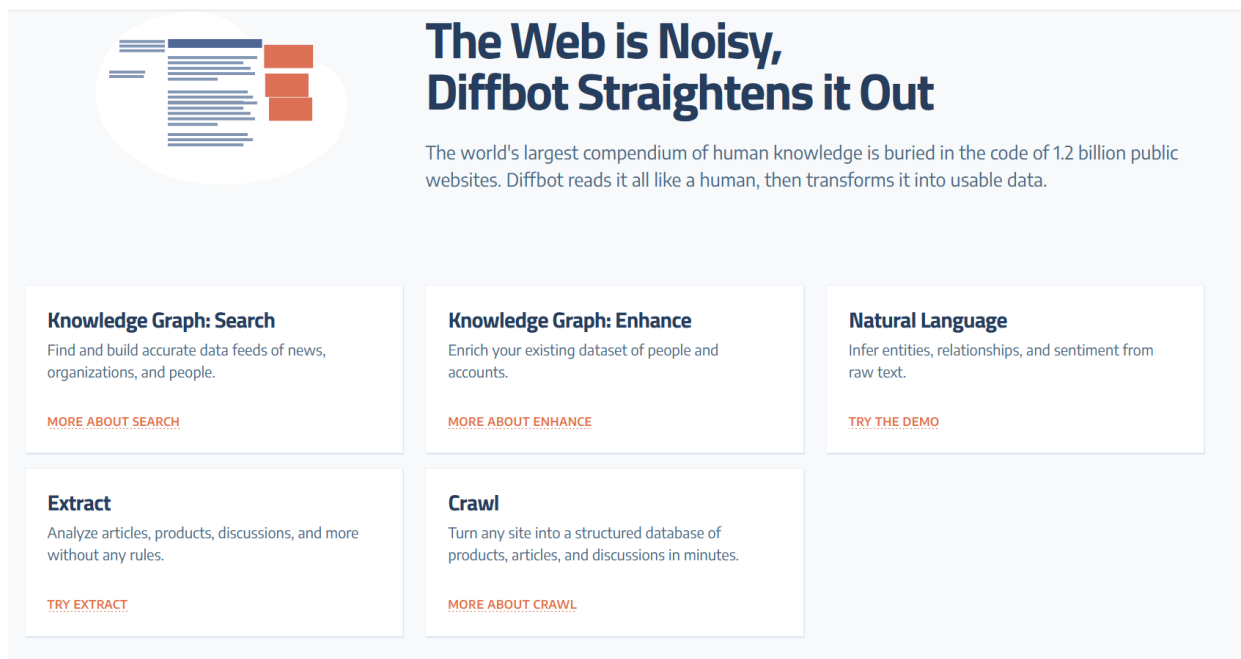


Рисунок 1.5 – Програмне забезпечення Diffbot [7], [16]

Серед особливостей сервісу Diffbot, що надає можливості для збирання даних з вебсайту, можна виділити такі [7], [16]:

- можливість збирання даних з вебсайтів через зручний інтерфейс;
- відсутність необхідності навичок програмування у користувачів;
- можливість вилучення довільного елемента HTML;
- видобування списків вебсторінок і вилучення відповідних детальних сторінок;
- можливість оброблення великих обсягів даних;
- можливість застосування регулярних виразів;

- засоби API;
- ефективна служба підтримки;
- засоби збереження даних у форматі CSV, XML, HTML тощо;
- онлайн-сховище даних;
- використання хмарних технологій.

Серед недоліків програмного забезпечення Diffbot виділяється його відносна велика вартість та відсутність безкоштовного тарифного плану [7], [16]. Крім того, відсутня україномовна версія цього програмного забезпечення [7], [16].

Порівняльну характеристику програмного забезпечення для збирання даних з вебсайтів наведено у таблиці 1.1.

Таблиця 1.1 – Порівняння програмного забезпечення для збирання даних з вебсайтів

Критерій порівняння	Octoparse [12]	import.io [13]	Mozenda [14]	ParseHub [15]	Diffbot [16]
Використання хмарних обчислень	+	–	+	+	+
Засоби API	+	–	+	+	+
Наявність безкоштовного тарифного плану	+	+	–	+	–
Зручність інтерфейсу	+	+	+–	+	+
Україномовна версія	–	–	–	–	–
Онлайн-сховище даних	–	+	+	–	+

За результатами проведеного аналізу можна зробити висновок, що у наш час існує досить багато програмних систем для збирання даних з вебсайтів. Проте деякі з них є іноді можуть некоректно збирати дані з вебсайтів, інші є досить дорогими, можуть містити багато функцій, що часто є зайвими та ускладнюють роботу користувача з програмою. Крім того,

універсальні програми збирання даних можуть мати проблеми, пов'язані зі заборонаю або обмеженням доступу, оскільки деякі вебсайти можуть мати правила, які обмежують доступ або забороняють програмам збирати їх дані. Це може призвести до блокування програми або зниження її продуктивності. Також існують проблеми з обмеженням на кількість запитів, низькою якістю даних, а також зі зміною структури вебсайту, внаслідок чого програма збирання даних може не зможти знайти та зібрати необхідні дані.

Тому для аналізу конкретного вебсайту актуальною є розробка програмного забезпечення для збирання даних.

При розробці програми для збирання даних з вебсайтів необхідно забезпечити такі функціональні вимоги:

- автоматичне генерування посилань на сторінки вебсайту, з яких будуть видобуватися дані;
- робота з чергою на парсинг даних;
- можливість видобування переліку телефонів з визначеного вебсайту;
- збирання даних назв підприємств;
- можливість збирання поштових адрес;
- можливість збирання електронних адрес;
- збереження зібраних та систематизованих даних у базу даних.

1.3 Висновки за розділом 1

Визначено, що у наш час широко застосовуються засоби вебскрапінгу, тобто збирання даних з вебсайтів, що дозволяє видобувати та систематизувати велику кількість вебданих, виконувати вебіндексування, аналіз даних, моніторинг змін цін і порівняння цін, збирання оглядів продуктів, виконувати збір списків нерухомості, моніторинг погодних даних, виявлення змін вебсайтів, відстеження присутності та репутації в Інтернеті, а також розв'язувати інші завдання. Виконано класифікацію програм видобування даних з вебсайтів. Проаналізовано галузі використання програм для збирання

даних з вебсайтів. Наведено етапи узагальненого процесу видобування даних з вебсайтів. Визначено фактори, які необхідно врахувати у процесі розроблення програмного забезпечення для збирання даних з вебсайтів.

За результатами провередного аналізу зроблено висновок про те, що наш час існує досить багато програмних систем для збирання даних з вебсайтів. Проте деякі з них є іноді можуть некоректно збирати дані з вебсайтів, інші є досить дорогими, можуть містити багато функцій, що часто є зайвими та ускладнюють роботу користувача з програмою. Крім того, універсальні програми збирання даних можуть мати проблеми, пов'язані зі заборною або обмеженням доступу, оскільки деякі вебсайти можуть мати правила, які обмежують доступ або забороняють програмам збирати їх дані. Це може призвести до блокування програми або зниження її продуктивності. Також існують проблеми з обмеженням на кількість запитів, низькою якістю даних, а також зі зміною структури вебсайту, внаслідок чого програма збирання даних може не змогти знайти та зібрати необхідні дані. Тому для аналізу конкретного вебсайту актуальною є розробка програмного забезпечення для збирання даних.

Сформульовано функціональні вимоги до програмного забезпечення для збирання даних з вебсайтів.

2 МАТЕРІАЛИ І МЕТОДИ

2.1 Вибір мови програмування та інших технологій розробки

При розробці програмного забезпечення для збирання даних з вебсайту було використано мову програмування Python [17]-[19].

Мова програмування Python є однією з найбільш популярних мов програмування в світі. Вона базується на об'єктно-орієнтованому підході, що дозволяє створювати програми швидко і ефективно. Python має широкий набір інтерфейсів для системних викликів та бібліотек операційних систем, що дозволяє розширювати можливості мови до C або C++ [17]-[19].

Python знаходить застосування у багатьох великих компаніях, таких як NASA, Google, YouTube, BitTorrent і багатьох інших. Вона використовується для розробки вебдодатків, створення наукових обчислень, аналізу даних, створення ігор, робототехніки та багато іншого [17]-[19].

Однією з переваг Python є його простота та легкість в навчанні. Він має читабельний та зрозумілий синтаксис, що дозволяє програмістам швидко вивчати мову та писати код. Крім того, він є крос-платформним, що дозволяє розробляти програми на будь-якій операційній системі [17]-[19].

Python також має велику кількість бібліотек та модулів, що спрощує процес розробки. Він підтримує різні парадигми програмування, включаючи процедурне, об'єктно-орієнтоване та функціональне програмування, що дозволяє програмістам використовувати найбільш зручний для них підхід [17]-[19].

Python — це високорівнева, інтерпретована, інтерактивна та об'єктно-орієнтована мова сценаріїв. Python створений для високої читабельності. Він часто використовує англійські ключові слова там, де інші мови використовують знаки пунктуації, і має менше синтаксичних конструкцій, ніж інші мови [17]-[19].

Python є інтерпретованою мовою програмування, тому Python обробляється під час виконання інтерпретатором. Не потрібно компілювати

програму перед її виконанням. Python є інтерактивним, розробник може використовувати підказки і взаємодіяти з інтерпретатором безпосередньо, щоб написати свої програми [17]-[19].

Python є об'єктно-орієнтованим, підтримує об'єктно-орієнтований стиль або техніку програмування, яка інкапсулює код всередині об'єктів. Завдяки своїй простоті та легкості в освоєнні, Python дозволяє створювати прототипи програм швидко та ефективно. Мова програмування Python має багато інтерфейсів для системних викликів та бібліотек ОС, що дозволяє розширювати можливості програми [17]-[19].

Python є зручною мовою для початку розробки, яка підтримує розробку широкого спектру додатків від простої обробки тексту до веббраузерів та ігор [17]-[19].

Мова програмування Python є простою в освоєнні, має кілька ключових слів, просту структуру та чітко визначений синтаксис. Це дозволяє швидко опанувати мову. Код Python більш чітко визначений і видимий для очей у порівнянні з кодом деяких інших мов програмування [17]-[19].

Python є простим в обслуговуванні, тобто вихідний код Python досить простий у обслуговуванні. Python має дуже потужну стандартну бібліотеку, яка портативною та сумісною з різними платформами в UNIX, Windows і Macintosh [17]-[19].

Python підтримує інтерактивний режим, який дозволяє інтерактивне тестування та налагодження фрагментів коду [17]-[19].

Python може працювати на різноманітних апаратних платформах і має однаковий інтерфейс на всіх платформах. Розробник може додавати модулі низького рівня до інтерпретатора Python. Ці модулі дозволяють програмістам додавати або налаштовувати свої інструменти, щоб вони були більш ефективними [17]-[19].

Python надає інтерфейси для всіх основних комерційних баз даних. Python підтримує програми графічного інтерфейсу користувача, які можна створювати та переносити на багато системних викликів, бібліотек і систем

Windows, таких як Windows MFC, Macintosh і система X Window Unix. Python забезпечує кращу структуру та підтримку для великих програм, ніж сценарії оболонки [17]-[19].

Мова програмування Python підтримує функціональні та структуровані методи програмування, а також об'єктно-орієнтоване програмування. Його можна використовувати як мову сценаріїв або скомпільувати в байт-код для створення великих програм. Він забезпечує динамічні типи даних дуже високого рівня та підтримує динамічну перевірку типів.

Мова програмування Python підтримує автоматичний збір сміття. Код на мові програмування Python можна легко інтегрувати з C, C++ та Java.

Розглянемо інструментальні засоби, які можна використовувати для розроблення програмного забезпечення збирання даних з вебсайтів [1]-[6]. Відзначимо, що загалом для вебсерфінгу потрібне знання мов програмування, досить зручною з яких для розв'язання подібних завдань є Python [5]. Python постачається з досить великою кількістю вбудованих бібліотек, які спрощують пошук аналіз вебсторінок. Серед бібліотек, які можуть бути використані для збирання даних з вебсторінок можна відзначити такі [5]:

- BeautifulSoup – широко використовувана бібліотека Python для парсингу даних із документів XML і HTML. Упорядковуючи вміст у простому дереві, BeautifulSoup дозволяє дуже легко знаходити кілька розділів інформації та керувати ними. Це інструмент аналізу великих даних [5];

- Scrapy – це фреймворк на основі Python, який отримує та витягує структуровані дані з Інтернету. В основному використовується для збору, обробки та архівування даних. Його можна використовувати як засіб для збирання даних з вебсторінок, а також для отримання даних через API [5];

- Pandas – універсальна бібліотека Python, яка використовується для обробки та представлення даних. Цю бібліотеку можна використовувати разом з BeautifulSoup для видобування та аналізу даних з вебсайтів. Основна перевага використання Pandas полягає в тому, що аналітики можуть

виконувати весь процес аналізу даних однією мовою (не потрібно перемикатися на іншу мову, наприклад R) [5].

Узагалі, Python є потужною мовою програмування, що має широкі можливості та знаходить застосування у багатьох галузях. Його простота та ефективність роблять його популярним вибором для програмістів усього світу [17]-[19].

Обґрунтування вибору мови програмування для розробки програмного забезпечення обміну зображеннями наведено у таблиці 2.1.

Таблиця 2.1 – Обґрунтування вибору мови програмування для розробки програмного забезпечення збирання даних з вебсайту

Критерій порівняння мов програмування	Мова програмування		
	Python	C#	Java
Наявність бібліотек для аналізу вебданих	+	+-	+
Зручність для створення програмного забезпечення збирання даних з вебсайту	+	+-	+-
Портативність	+	+-	+
Інтерактивність	+	+	+-
Простота використання	+	+-	+-

Отже, для реалізації програмного забезпечення збирання даних з вебсайту обрано мову програмування Python, яка являє собою об'єктно-орієнтовану мову програмування високого рівня з вбудованими структурами даних і динамічною семантикою, підтримує кілька парадигм програмування, таких як структурне, об'єктно-орієнтоване та функціональне програмування, а також має великий набір бібліотек для аналізу вебданих.

2.2 Вибір середовища розробки для створення програмного забезпечення збирання даних з вебсайту

В якості середовища розробки для створення програмного забезпечення збирання даних з вебсайту було обрано Visual Studio Code [20], [21].

Visual Studio Code — це безкоштовне програмне забезпечення від Microsoft. Visual Studio Code доступний для Windows, Linux і macOS. Хоча редактор дуже простий, він містить потужні функції, які роблять Visual Studio Code одним із найпопулярніших інструментів середовища розробки останнім часом [20], [21].

Visual Studio Code підтримує широкий спектр мов програмування, від Java, C++ і Python, а також засоби CSS, Go і Dockerfile. Visual Studio Code також дозволяє додавати нові розширення, такі як кодовий лінтер, налагоджувач, хмарна та веброботка [20], [21].

Інтерфейс Visual Studio Code забезпечує більше взаємодії, ніж деякі інші редактори. Щоб спростити роботу з користувачем, код Visual Studio поділено на п'ять основних розділів: панель активності, бічна панель, групи редакторів, звичайна панель, рядок стану [20], [21].

Visual Studio Code — це безкоштовний, простий, але потужний засіб, який працює на робочому столі та в Інтернеті та доступний для Windows, macOS, Linux і Raspberry Pi. Він має вбудовану підтримку JavaScript, TypeScript, Node.js і підтримує інші мови програмування (C++, C#, Java, Python, PHP, Go тощо), середовища виконання (наприклад, .NET і Unity), середовища (Docker і Kubernetes тощо), хмарні сервіси (Amazon Web Services, Microsoft Azure, Google Cloud Platform тощо) [20], [21].

Окрім легкого та швидкого початку роботи, Visual Studio Code має повний код IntelliSense для змінних, методів та імпортованих модулів; стіл; праймінг, мультикурсорне редагування, відображення параметрів та інші потужні функції редагування; код ревізії; і керування вихідним кодом,

включаючи підтримку Git. Більшість із них походять із методів Visual Studio [20], [21].

Visual Studio Code створений за допомогою Electron Shell, Node.js, TypeScript і Proctor і оновлюється щомісяця. За потреби вносяться додаткові оновлення. Різноманітність довідки залежить від різних мов програмування та розширень: від простого підсвічування рядків і зіставлення дужок до обробки та рекурсії. Якщо у розробника немає мовного сервера, ви можете додати рідну підтримку для потрібної мови за допомогою калькулятора TextMate [20], [21].

Код у репозиторії коду Visual Studio є відкритим вихідним кодом. Продукти Visual Studio Code містять лише невеликий відсоток власного вмісту Microsoft, тому на них поширюється стандартна ліцензія Microsoft. Він має комерційну ліцензію, але є безкоштовним [20], [21].

У Visual Studio Code мільйони активних користувачів. Багато користувачів Visual Studio Code є розробниками Google або Facebook. Більшість користувачів Visual Studio Code є дослідниками даних, які віддають перевагу повній підтримці Python або R під час використання ноутбуків Jupyter [20], [21].

Розробникам подобається легкість Visual Studio Code зі зручним редактором із пов'язаним синтаксисом, повним кодом, реактором коду, функціями налагодження та аудиту бази даних. Розробникам хмар і контейнерів подобаються можливості віддаленого керування Visual Studio Code і підтримка великих хмар, Docker і Kubernetes. Розробники працюють у командах, як інтеграція Git від Visual Studio Code [20], [21].

Visual Studio може створювати нові проєкти з шаблонів. Visual Studio Code не може цього зробити, але можна використовувати інструмент на панелі інструментів, наприклад Yeoman, щоб створити список проєктів у консолі Visual Studio Code. Visual Studio знадобиться небагато часу, щоб відкрити великі проєкти. Він також використовується для керування всіма драйверами.

Однак дисковий простір тепер більший, і тепер можна інстальювати лише необхідні компоненти Visual Studio [20], [21].

Обґрунтування вибору середовища розробки для створення програмного забезпечення збирання даних з вебсайтів наведено у таблиці 2.2.

Таблиця 2.2 – Обґрунтування вибору середовища розробки для створення програмного забезпечення збирання даних з вебсайтів

Критерій порівняння середовищ розробки	Середовище розробки		
	Visual Studio Code	Sublime Text	PyCharm
Безкоштовність	+	+	—
Візуальний редактор коду	+	—	+—
Зручність використання	+	+—	+—
Підтримка рефакторингу	+	—	+
Підтримка хмарних сервісів	+	—	—
Відкритість ПЗ	+	—	—

Таким чином, для створення програмного забезпечення збирання даних з вебсайтів обрано середовище розробки Visual Studio Code.

2.3 Висновки за розділом 2

Для створення програмного забезпечення збирання даних з вебсайтів обрано мову програмування Python, яка являє собою об'єктно-орієнтовану мову програмування високого рівня з вбудованими структурами даних і динамічною семантикою, підтримує кілька парадигм програмування, таких як

структурне, об'єктно-орієнтоване та функціональне програмування, а також має великий набір бібліотек для аналізу вебданих.

В якості середовища розробки для створення програмного забезпечення збирання даних з вебсайтів обрано Visual Studio Code, що є безкоштовним програмним забезпеченням, містить потужні функції та інструменти середовища розробки, підтримує широкий спектр мов програмування.

3 ОПИС ПРОГРАМИ

3.1 Структура процесів програми збирання даних з вебсайту

Відповідно до технічного завдання було обрано структуру програми збирання даних з вебсайту, яку можна представити у вигляді двох основних процесів та п'яти основних модулів. Структура процесів програми збирання даних з вебсайту зображена на рисунку 3.1.

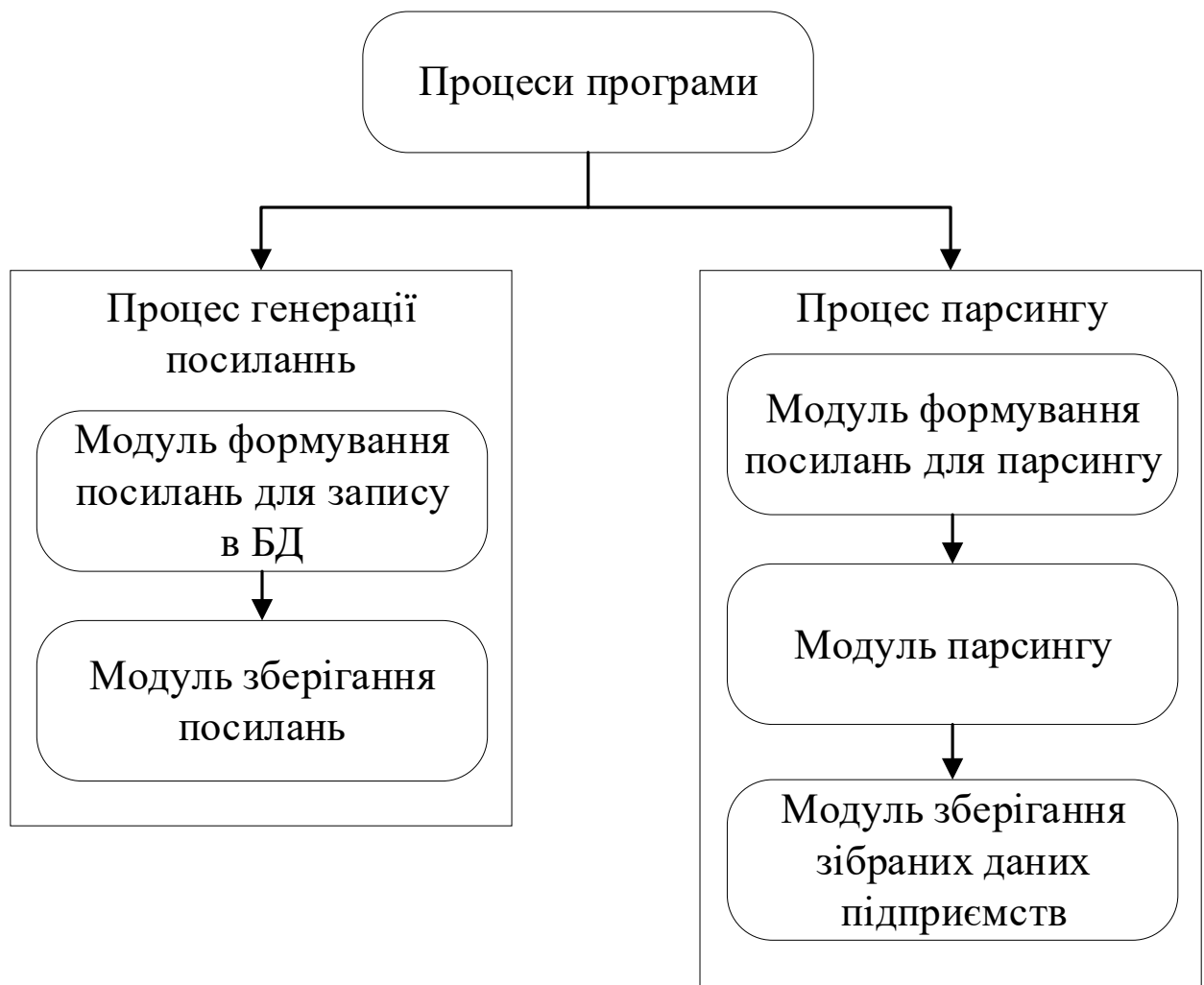


Рисунок 3.1 – Структура процесів програми збирання даних з вебсайту

Як видно з рис. 3.1 основними процесами програми збирання даних з вебсайту є процес генерації посилань та процес парсингу. Така структури системи дозволяє пришвидшити роботу програми збирання даних з вебсайту

та оптимально використовувати ресурси комп'ютера, завдяки паралельному виконанню двох основних процесів.

3.2 Опис програмних модулів процесу генерації посилань

Процес генерації посилань є процес, результатом якого є база даних заповнена посиланнями, які будуть аналізуватися в процесі парсингу.

Даний процес складається з двох модулів:

- модуль формування посилань для запису в базу даних;
- модуль зберігання посилань.

Модуль формування посилань для запису в базу даних відповідає за генерацію посилань за наступним правилом: у посилання є єдина структура «<http://www.icc.org/details/<number>.htm>», де на місці <number> стоїть номер конкретного підприємства на сайті [icc.com](http://www.icc.com). Емпіричним шляхом було виявлено, що номери знаходяться в інтервалі [1000, 16939579]. Після формування посилання, із нього формується спеціальне повідомлення, тобто дані упорядковуються у json-форматі та надсилається до черги на запис у базу даних.

Модуль зберігання посилань відповідає за зберігання посилань у базі даних. Даний модуль являє собою консьюмер, який раз на секунду надсилає запит та перевіряє наявність повідомлень у черзі на запис у базі даних. Якщо такі мають, то даний модуль їх оброблює та записує в базу даних.

3.3 Опис програмних модулів процесу парсингу

Процес парсингу є процес, результатом якого є БД, яка заповнена даними підприємств, такими як номери телефонів, електронні пошти, назви та адреси підприємств. Даний процес складається з наступних модулів:

- модуль формування посилань для парсингу;
- модуль парсингу;

– модуль зберігання зібраних даних підприємств.

Модуль формування посилань для парсингу відповідає за вибір посилань з БД, які необхідно надіслати до черги на парсинг. Після отримання списку посилань, із кожного з них формується спеціальне повідомлення, тобто дані упорядковуються у json-форматі, та надсилається до черги на парсинг.

Модуль парсингу відповідає за аналіз сторінок та вибірку даних підприємств, таких як номери телефонів, електронні пошти, назви та адреси підприємств. Після завершення аналізу сторінки та вибірки даних підприємств із них формується спеціальне повідомлення, тобто дані упорядковуються у json-форматі, та надсилається до черги на зберігання даних підприємства.

Модуль зберігання зібраних даних підприємств відповідає за зберігання даних підприємств у БД. Даний модуль являє собою консьюмер, який раз на секунду надсилає запит та перевіряє наявність повідомлень у черзі на запис у БД. Якщо такі мають, то даний модуль їх оброблює та записує в БД.

3.4 Архітектура програмного забезпечення для збирання даних з вебсайту

Для програмного забезпечення для збирання даних з вебсайту було спроектовано схему на основі основних програмних процесів, описаних у підрозділі 3.1. Дана схема зображена на рисунку 3.2.

На даній схемі зображені 9 програмних компонентів системи.

Модуль бази даних містить базу даних, яка зберігає інформацію про знайдені дані підприємств та посилань на сторінки сайту iss.com із описом підприємств.

Паблішер згенерованих посилань — це модуль, що відповідає за генерацію посилань сайту iss, та надсилання їх у json-форматі до черги для зберігання посилань за допомогою бібліотеки Pika.

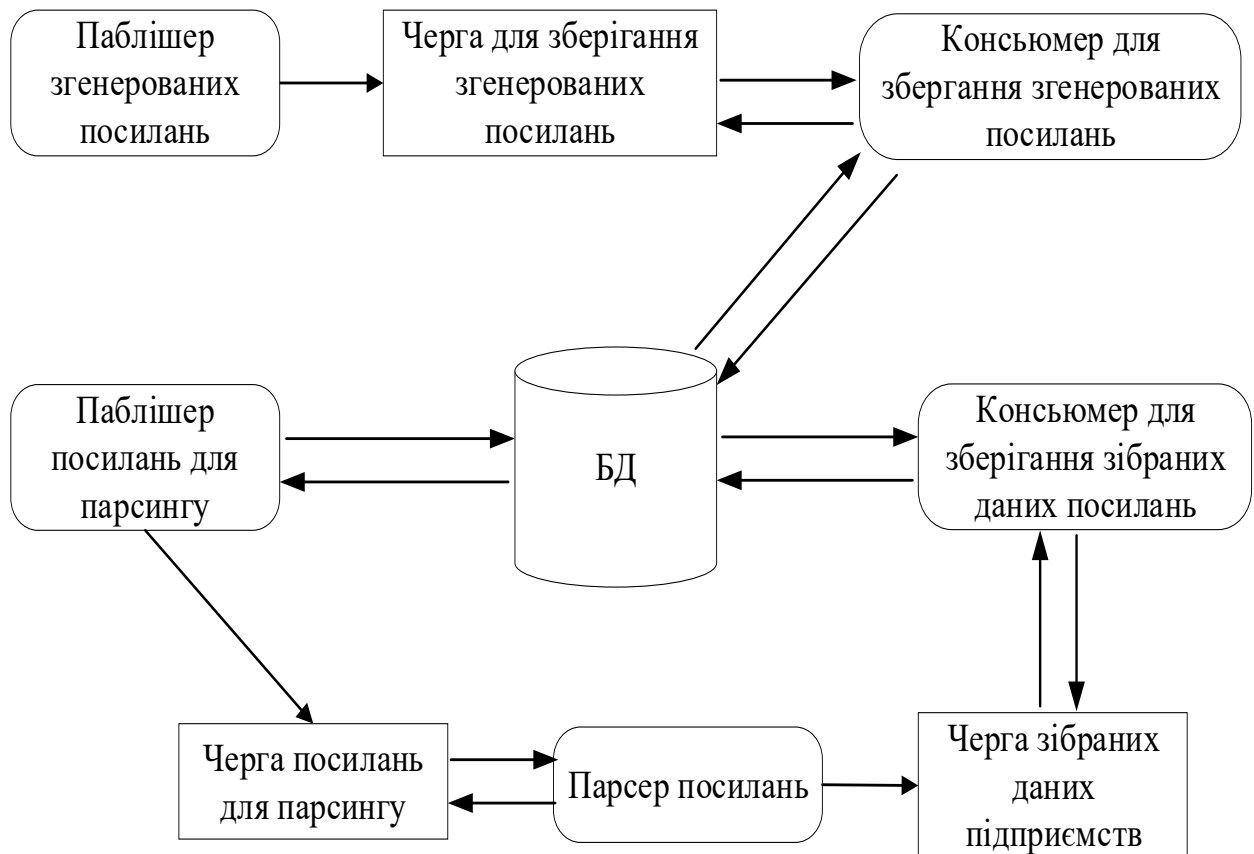


Рисунок 3.2 – Архітектура програмного забезпечення для збирання даних з вебсайту

Консьюмер для збергання згенерованих посилань – це модуль, що раз на секунду надсилає запит до черги для зберігання згенерованих посилань на наявність нових посилань за допомогою методів бібліотеки AMQP. Якщо у черзі є посилання, то даний модуль їх зберігає у буфері до тих пір, поки кількість посилань у буфері не буде дорівнювати п'ятсот одиниць та надсилає запит на зберігання до модуля БД у вигляді SQL-скрипту.

Паблішер посилань для парсингу – це модуль, що надсилає запити до БД на вибірку посилань, для парсингу. Якщо такі посилання є, то він їх надсилає до черги на парсинг у json-форматі за допомогою методів бібліотеки Pika.

Парсер посилань – це модуль, що надсилає запит до черги посилань для парсингу на наявність посилань. Якщо такі наявні, то даний модуль за даними посиланнями надсилає запит на сайт iss.com та отримує відповідь з

нього. Після отримання відповіді починається її обробка, та, за допомогою селекторів ХРАТН, починається збір даних підприємств, таких як номери телефонів, електронні пошти, назви та адреси підприємств. Після завершення обробки та збору даних підприємств даний модуль згруповує дані у json-форматі та надсилає їх до черги для зберігання даних підприємств за допомогою методів бібліотеки Riik.

Консьюмер для зберігання зібраних даних посилань – це модуль, що раз на секунду надсилає запит до черги для зберігання даних підприємств на наявність нових даних за допомогою методів бібліотеки AMQP. Якщо у черзі є дані, то даний модуль їх зберігає у буфері до тих пір, поки кількість посилань у буфері не буде дорівнювати п'ятсот одиниць та надсилає запит на зберігання до модуля БД у вигляді SQL-скрипту.

Черга посилань для парсингу, черга зібраних даних підприємств та черга для зберігання згенерованих посилань виступають у якості тимчасового сховища даних, які представлені у json-форматі.

3.5 Структура бази даних

Оскільки база даних повинна зберігати тільки дані про підприємства, тому можна обмежитись тільки однією таблицею, що зберігатиме дані. Структура бази даних у вигляді концептуальної моделі зображена на рисунку 3.3.

Оскільки у базі буди приблизно п'ять мільйонів записів, тому були додані індекси для полів, таких як дата створення, надсилання замовнику та номер телефону, для прискорення обробки запитів базою даних. Індекси необхідні для того, щоб база даних використовувала для вибірки значень спеціальні алгоритми фільтрації та пошуку.



Рисунок 3.3– Концептуальна модель бази даних

3.6 Файлова структура проєкту програмного забезпечення для збирання даних з вебсайту

Головна структура проєкту програмного забезпечення для збирання даних з вебсайту зображена на рисунку 3.4.

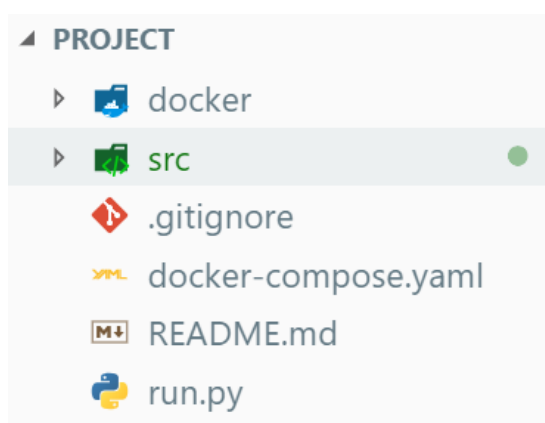


Рисунок 3.4 – Головна структура проєкту програмного забезпечення для збирання даних з вебсайту

Головна структура проєкту програмного забезпечення для збирання даних з вебсайту складається з двох директорій та чотирьох файлів.

Файл «.gitignore» містить у собі інструкції для Git, для фільтрування файлів, що містимуться у репозиторії.

Файл «docker-compose.yaml» містить у собі головні інструкції для налаштування Docker-середовища програмного забезпечення для збирання даних з вебсайту.

Файл «Readme.md» містить у собі скорочений набір інструкцій та пояснень для проєкту програмного забезпечення для збирання даних з вебсайту.

Файл «run.py» являє собою основним входом програмного забезпечення для збирання даних з вебсайту, тобто утилітою для запуску та маніпулювання станами проєкту.

Директорія «docker» – директорія, яка містить у собі файли налаштування для Docker-оточення програмного забезпечення для збирання даних з вебсайту. На рисунку 3.5 зображена структура директорії «docker».

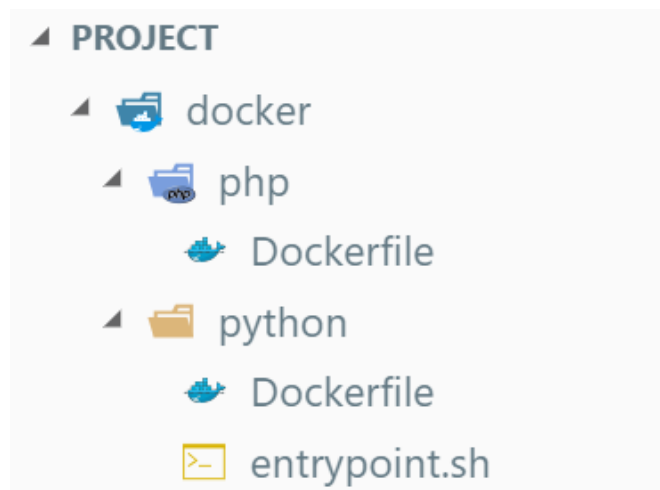


Рисунок 3.5 – Файлова структура директорії «docker» програмного забезпечення для збирання даних з вебсайту

Файл «Dockerfile» (рис. 3.5) містить у собі конфігурації для конкретного контейнеру, в даному випадку для контейнерів із php та python,

які використовуються як основи для запуску парсерів, консьюмерів та паблішерів.

Директорія «src» (рис.3.4) містить у собі файли та директорії самого проєкту програмного забезпечення для збирання даних з вебсайту. Структура директорії «src» зображена на рисунку 3.6.

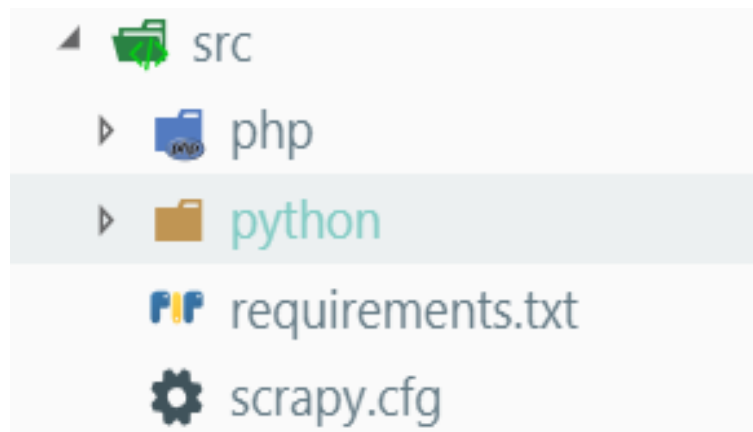


Рисунок 3.6 – Файлова структура директорії «src» програмного забезпечення для збирання даних з вебсайту

Директорія «src» (рис. 3.6) програмного забезпечення для збирання даних з вебсайту включає в себе такі файли та директорії:

- директорія «php» – директорія, що містить у собі консьюмери, написні мовою програмування PHP;
- директорія «python» містить у собі парсери, написані мовою програмування Python;
- файл «requirements.txt» – містить у собі список бібліотек, використаних у розробці проєкту;
- файл «scrapy.cfg» – файл конфігурацій для парсерів.

Директорія «python», що знаходиться у папці «src», складається з чотирьох піддиректорій та п'яти файлів. Структура директорії зображена на рисунку 3.7.

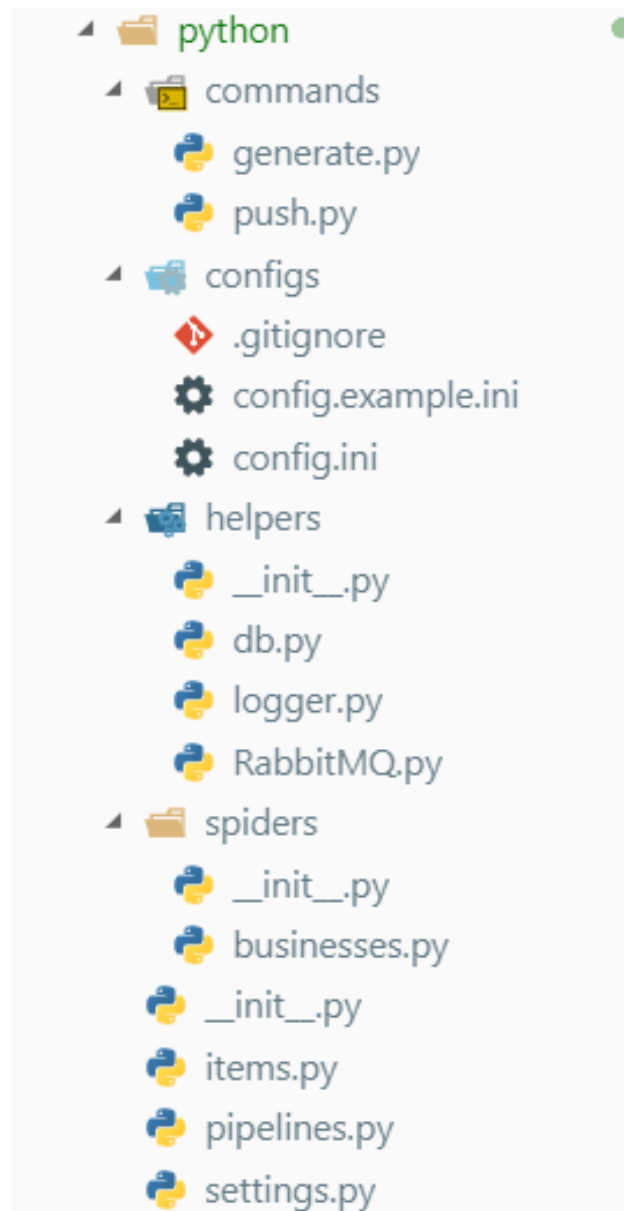


Рисунок 3.7 – Файлова структура директорії «python» програмного забезпечення для збирання даних з вебсайту

У даних директоріях програмного забезпечення для збирання даних з вебсайту присутні файли «__init__.py». Це файли, що вказують мові програмування Python на те, що директорія, в котрій розташований даний файл є пакетом, тобто функціонал з нього можна використовувати і в інших директоріях.

Директорія «commands» програмного забезпечення для збирання даних з вебсайту містить у собі команди для роботи системи, що зберігаються у таких файлах:

- «generate.py» – файл, у якому реалізовано команду, після виклику якої починається генерація посилань для парсингу;

- «push.py» – файл, у якому реалізовано команду, після виклику якої починається формування з посилань з БД спеціальні повідомлення у json-формат, та надсилення їх до черги на парсинг.

Директорія «configs» містить у собі файл із списком конфігурацій для парсингу «config.ini». У даному файлі прописані такі параметри, як дані для підключення до СУБД MySQL та черги RabbitMQ, назви черг, у яких знаходитимуться посилання для парсингу та забрані дані підприємств.

Директорія «helpers» програмного забезпечення для збирання даних з вебсайту містить у собі файли, у котрих реалізовано:

- у файлі «db.py» – підключення до СУБД MySQL;
- у файлі «logget.py» – реалізовано функціонал, що забезпечує системі структуровані повідомлення під час роботи програми;
- у файлі «RabbitMQ.py» – підключення до черги RabbitMQ.

У директорії «spiders» програмного забезпечення для збирання даних з вебсайту розташовано файл «business.py», який містить у собі функціонал, що реалізує роботу парсеру.

У файлі «items.py» реалізована спеціальна сутність підприємства для роботи парсеру.

У файлі «pipelines.py» реалізована функціонал, що оброблює зібрані дані та надсилає їх до черги на зберігання даних.

У файлі «settings.py» знаходяться основні налаштування парсерів.

3.7 Висновки за розділом 3

Розроблено програмне забезпечення для збирання даних з вебсайту. Запропоновано структуру процесів програми збирання даних з вебсайту. Визначено, що основними процесами програми є процес генерації посилань та процес парсингу. Процес генерації посилань є процес, результатом якого є

база даних заповнена посиланнями, що будуть аналізуватися в процесі парсингу. Цей процес реалізується модулем формування посилань для запису в базу даних та модулем зберігання посилань. Процес парсингу є процес, результатом якого є база, заповнена даними підприємств. Даний процес реалізується за допомогою модулю формування посилань для парсингу, модулю парсингу та модулю зберігання зібраних даних підприємств.

Розроблено архітектуру програмного забезпечення для збирання даних з вебсайту. Описано файлову структуру проєкту програмного забезпечення для збирання даних з вебсайту.

4 ЕКСПЛУАТАЦІЯ, ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПРОГРАМИ

4.1 Призначення й умови застосування програми

Програма призначена для збирання даних з вебсайту. Програма написана на мові Python, в якості середовища розробки для створення програмного забезпечення збирання даних з вебсайтів обрано Visual Studio Code, що є безкоштовним програмним забезпеченням, містить потужні функції та інструменти середовища розробки, підтримує широкий спектр мов програмування.

Програмне забезпечення для збирання даних з вебсайту забезпечує виконання таких функцій:

- автоматичне генерування посилань на сторінки вебсайту, з яких будуть видобуватися дані;
- робота з чергою на парсинг даних;
- можливість видобування переліку телефонів з визначеного вебсайту;
- збирання даних назв підприємств;
- можливість збирання поштових адрес;
- можливість збирання електронних адрес;
- збереження зібраних та систематизованих даних у базу даних.

Для експлуатації програмного забезпечення для збирання даних з вебсайту необхідно таке апаратне та програмне забезпечення:

- процесор з тактовою частотою 1ГГц або вище;
- оперативна пам'ять – 1Гб або вище;
- операційна система Windows 10 Pro або будь-який дистрибутив Linux;
- засоби доступу до Інтернету.

Функціональні характеристики розробленого програмного забезпечення для збирання даних з вебсайту наведено у технічному завданні

(додаток А). Фрагмент тексту програмного забезпечення для збирання даних з вебсайту наведено у додатку Б.

4.2 Характеристики програми та її встановлення

Програмне забезпечення для збирання даних з вебсайту працює відповідно до потреб користувача.

Для встановлення програми необхідно виконати такі дії:

- перевірити наявність встановлених у системі python3, docker та git;
- відкрити консоль у директорії, у якій знаходитимуться файли проєкту;
- у консолі виконати команду «git clone path project», де path – шлях до проєкту у розподіленій системі управління версіями;
- у консолі виконати команду cd «./project/»;
- у завантаженому проєкті відкрити файл docker-compose.yaml, та замінити конфігурації підключення до СКБД MySQL та черги RabbitMQ;
- зробити копії таких файлів, як src/php/_config/.env.example та src/python/configs/config.ini та назвати їх відповідно .env та config.ini;
- у вище створених файлах вписати у відповідних полях дані конфігурацій підключення до бази даних та черги, що вказані в файлі docker-compose.yaml;
- у консолі виконати команду «python run.py start».

4.3 Інструкція по експлуатації програми

4.3.1 Звернення до програми

Для запуску програми користувачу у консолі необхідно за допомогою команди python запустити відповідно програму. Це можна зробити за допомогою команди python prog.py, де prog.py – назва програмного забезпечення для збирання даних з вебсайту.

4.3.2 Вхідні й вихідні дані

Вхідними даними до програмного забезпечення для збирання даних з вебсайту є інформація про сайт, з якого необхідно збирати та систематизувати інформацію.

Вихідними даними програмного забезпечення для збирання даних з вебсайту є автоматично згенеровані посилання на сторінки вебсайту, з яких будуть видобуватися дані, а також відібрана інформація (перелік телефонів, назви підприємств, поштові адреси, електронні адреси та ін.).

4.3.3 Повідомлення

Користувач програмного забезпечення для збирання даних з вебсайту може отримати такі повідомлення: повідомлення про успішне збирання даних, повідомлення про неможливість збору даних у зв'язку з відсутністю доступу до вказаного вебсайту та ін.

4.4 Виконання програмного забезпечення для збирання даних з вебсайту

Оскільки програмне забезпечення для збирання даних з вебсайту являє собою консольну програму, то для її запуску користувачу спочатку необхідно запустити консоль (наприклад, в операційній системі Windows, за допомогою команди PowerShell). Після цього у консолі виконати команду `python prog.py`, де `prog.py` – назва програмного забезпечення для збирання даних з вебсайту.

Після запуску програма за необхідності автоматично встановить потрібні модулі для роботи у спеціальні контейнери. Приклад встановлення необхідних модулів зображено на рис. 4.1.

```

docker-compose up --build
pulling composer (composer:latest)...
latest: Pulling from library/composer
bdf0201b3a05: Pull complete
a8a561c50e0: Pull complete
f581634c6ada: Pull complete
f205a7399250: Pull complete
e5d36c3313bd: Pull complete
dda578191749: Pull complete
9dfcb7436856: Pull complete
7354b0c6b605: Pull complete
9d9c6c52b0b9: Pull complete
fd36bde9cd99: Pull complete
f517c9d6f096: Pull complete
5bb1f6497eb1: Pull complete
e295eada078a: Pull complete
Building app_php_business_saver
Step 1/5 : FROM php:7.2-fpm
7.2-fpm: Pulling from library/php
27833a3ba0a5: Already exists
2d79f6773a3c: Pull complete
f5dd9a448b82: Pull complete
95719a57e42b: Pull complete
cb5feddf5d9a: Pull complete
b0bc56502884: Pull complete
ef367063990b: Pull complete
c44248836afe: Pull complete
8233ff6e403: Pull complete
b7341166f141: Pull complete
Digest: sha256:56488a6f99763c24f9eeae0ffa5fbfb6f81dc5217587d4da301b49016a6
Status: Downloaded newer image for php:7.2-fpm
--> 53313db4f430
Step 2/5 : RUN apt-get update && apt-get install -y libfreetype6-dev libjpeg62-turbo-dev libpng-dev && docker-php-ext-install -j$(nproc) iconv
--> Running in 3bbc41949302
Get:1 http://security-cdn.debian.org/debian-security stretch/updates InRelease [94.3 kB]
Get:2 http://security-cdn.debian.org/debian-security buster/updates InRelease [39.1 kB]
Get:3 http://cdn-fastly.deb.debian.org/debian stretch/updates InRelease [91.0 kB]
Get:4 http://cdn-fastly.deb.debian.org/debian stretch/main amd64 Packages [487 kB]
Get:5 http://cdn-fastly.deb.debian.org/debian-security stretch/updates/main amd64 Packages [46.8 kB]
Get:6 http://cdn-fastly.deb.debian.org/debian-security buster/updates/main amd64 Packages [1132 B]
Get:7 http://cdn-fastly.deb.debian.org/debian stretch/main amd64 Packages [31.7 kB]
Get:8 http://cdn-fastly.deb.debian.org/debian stretch/updates/main amd64 Packages [118 kB]
Get:9 http://cdn-fastly.deb.debian.org/debian buster/main amd64 Packages [7908 kB]
Get:10 http://cdn-fastly.deb.debian.org/debian stretch/main amd64 Packages [7082 kB]
Get:11 http://cdn-fastly.deb.debian.org/debian stretch/updates/main amd64 Packages [7082 kB]
Get:12 http://cdn-fastly.deb.debian.org/debian stretch/main amd64 Packages [7082 kB]
Get:13 http://cdn-fastly.deb.debian.org/debian stretch/main amd64 Packages [7082 kB]
Fetched 16.1 MB in 6s (2373 kB/s)
Reading package lists...
Building dependency tree...

```

Рисунок 4.1 – Приклад встановлення модулів на першому етапі роботи програми для збирання даних з вебсайту

Після встановлення необхідних модулів програмне забезпечення почне запускати окремі контейнери та видобувати дані з вебсайту, виводячи при цьому у консоль службову інформацію. Приклад роботи програми у процесі збирання даних з вебсайту наведено на рис. 4.2.

```

mysql_container MySQL init process done. Ready for start up.
mysql_container 05-06T21:14:11.278742Z 0 [Warning] TIMESTAMP with implicit DEFAULT value is deprecated. Please use --explicit_defaults_for_timestamp server option (see documentation for more details).
mysql_container 05-06T21:14:11.280087Z 0 [Note] mysqld (mysqld 5.7.26) starting as process 1...
mysql_container 05-06T21:14:11.284862Z 0 [Note] InnoDB: PUNCH HOLE support available
mysql_container 05-06T21:14:11.284862Z 0 [Note] InnoDB: Mutexes and rw_locks use GCC atomic builtins
mysql_container 05-06T21:14:11.284891Z 0 [Note] InnoDB: Uses event mutexes
mysql_container 05-06T21:14:11.284894Z 0 [Note] InnoDB: GCC builtin __atomic_thread_fence() is used for memory barrier
mysql_container 05-06T21:14:11.284897Z 0 [Note] InnoDB: Compressed tables use zlib 1.2.11
mysql_container 05-06T21:14:11.284900Z 0 [Note] InnoDB: Using Linux native AIO
mysql_container 05-06T21:14:11.285123Z 0 [Note] InnoDB: Number of pools: 1
mysql_container 05-06T21:14:11.285225Z 0 [Note] InnoDB: Using CPU crc32 instructions
mysql_container 05-06T21:14:11.285931Z 0 [Note] InnoDB: Initializing buffer pool, total size = 128M, instances = 1, chunk size = 128M
mysql_container 05-06T21:14:11.284926Z 0 [Note] InnoDB: Completed initialization of buffer pool
mysql_container 05-06T21:14:11.297412Z 0 [Note] InnoDB: If the mysqld execution user is authorized, page cleaner thread priority can be changed. See the man page of setpriority().
mysql_container 05-06T21:14:11.309115Z 0 [Note] InnoDB: Highest supported file format is Barracuda.
mysql_container 05-06T21:14:11.419437Z 0 [Note] InnoDB: Creating shared tablespace for temporary tables
mysql_container 05-06T21:14:11.419539Z 0 [Note] InnoDB: Setting file './ibtmp1' size to 12 MB. Physically writing the file full; Please wait ...
mysql_container 05-06T21:14:11.619855Z 0 [Note] InnoDB: File './ibtmp1' size is now 12 MB.
mysql_container 05-06T21:14:11.620966Z 0 [Note] InnoDB: 96 redo rollback segment(s) found. 96 redo rollback segment(s) are active.
mysql_container 05-06T21:14:11.620984Z 0 [Note] InnoDB: 32 non-redo rollback segment(s) are active.
mysql_container 05-06T21:14:11.621448Z 0 [Note] InnoDB: 5.7.26 started; log sequence number 12360578
mysql_container 05-06T21:14:11.621810Z 0 [Note] Plugin 'FEDERATED' is disabled.
mysql_container 05-06T21:14:11.625724Z 0 [Note] Found ca.pem, server-cert.pem and server-key.pem in data directory. Trying to enable SSL support using them
mysql_container 05-06T21:14:11.625872Z 0 [Warning] CA certificate ca.pem is self signed.
mysql_container 05-06T21:14:11.627180Z 0 [Note] InnoDB: Loading buffer pool(s) from /var/lib/mysql/ib_buffer_pool
mysql_container 05-06T21:14:11.630832Z 0 [Note] Server hostname (bind-address): '::', port: 3306
mysql_container 05-06T21:14:11.630855Z 0 [Note] IPv6 is available.
mysql_container 05-06T21:14:11.630665Z 0 [Note] - '::' resolves to '::';
mysql_container 05-06T21:14:11.631249Z 0 [Note] Server socket created on IP: '::'
mysql_container 05-06T21:14:11.632889Z 0 [Note] InnoDB: Buffer pool(s) load completed at 190506 21:14:11
mysql_container 05-06T21:14:11.742233Z 0 [Warning] Insecure configuration for --pid-file: Location '/var/run/mysqld' in the path is accessible to all OS users. Consider choosing a different directory.
mysql_container 05-06T21:14:11.750997Z 0 [Note] Event Scheduler: Loaded 0 events
mysql_container 05-06T21:14:11.751207Z 0 [Note] mysqld: ready for connections.
mysql_container Version: '5.7.26' socket: '/var/run/mysqld/mysqld.sock' port: 3306 MySQL Community Server (GPL)
mysql_container
mysql_container =INFO REPORT==== 6-May:21:14:21 ===
mysql_container accepting AMQP connection <0.1005.0> (172.22.0.4:51858 -> 172.22.0.3:5672)
mysql_container
mysql_container =INFO REPORT==== 6-May:21:14:30 ===
mysql_container accepting AMQP connection <0.1085.0> (172.22.0.5:39290 -> 172.22.0.3:5672)
mysql_container 2019-05-06T21:14:42.947269Z 4 [Note] Aborted connection 4 to db: 'icc' user: 'user' host: '172.22.0.5' (Got an error reading communication packet)
mysql_container
mysql_container =WARNING REPORT==== 6-May:21:14:42 ===
mysql_container closing AMQP connection <0.1085.0> (172.22.0.5:39290 -> 172.22.0.3:5672):
mysql_container client unexpectedly closed TCP connection
mysql_container
mysql_container =INFO REPORT==== 6-May:21:14:48 ===
mysql_container accepting AMQP connection <0.1249.0> (172.22.0.5:39296 -> 172.22.0.3:5672)

```

Рисунок 4.2 – Приклад роботи програми у процесі збирання даних з вебсайту

До виконання безпосередньо видобування даних з вебсайту програма виконує генерацію посилань на вебсторінки, доступні на вебсайті, зміст якого аналізується. Приклад згенерованих посилань зображено на рисунку 4.3.

iccbusinesses: 4,482 rows total (approximately), limited to 1,000

id	url	name	address_locality	address_region	postal_code	street_address	phone	clear_phone	fax	website	category	sent_to_customer	parse_status
4	http://www.icc.org/details/1003.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
5	http://www.icc.org/details/1004.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
6	http://www.icc.org/details/1005.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
7	http://www.icc.org/details/1006.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
8	http://www.icc.org/details/1007.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
9	http://www.icc.org/details/1008.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
10	http://www.icc.org/details/1009.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
11	http://www.icc.org/details/1010.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
12	http://www.icc.org/details/1011.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
13	http://www.icc.org/details/1012.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
14	http://www.icc.org/details/1013.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
15	http://www.icc.org/details/1014.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
16	http://www.icc.org/details/1015.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
17	http://www.icc.org/details/1016.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
18	http://www.icc.org/details/1017.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
19	http://www.icc.org/details/1018.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
20	http://www.icc.org/details/1019.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
21	http://www.icc.org/details/1020.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
22	http://www.icc.org/details/1021.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
23	http://www.icc.org/details/1022.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
24	http://www.icc.org/details/1023.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
25	http://www.icc.org/details/1024.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
26	http://www.icc.org/details/1025.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
27	http://www.icc.org/details/1026.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
28	http://www.icc.org/details/1027.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
29	http://www.icc.org/details/1028.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
30	http://www.icc.org/details/1029.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
31	http://www.icc.org/details/1030.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
32	http://www.icc.org/details/1031.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
33	http://www.icc.org/details/1032.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
34	http://www.icc.org/details/1033.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
35	http://www.icc.org/details/1034.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
36	http://www.icc.org/details/1035.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1

Filter: Regular expression

```

158 SHOW TABLE STATUS LIKE 'businesses';
159 SELECT 'id', 'url', 'name', 'address_locality', 'address_region', 'postal_code', 'street_address', 'phone', 'clear_phone', 'fax', LEFT('website', 256), 'category', 'sent_to_customer', 'parse_status', 'created'
160 SHOW CREATE TABLE 'icc'.businesses;
161 SHOW TABLE STATUS LIKE 'businesses';

```

icc: 2 tables 1:1 Connected: 00:02 h MariaDB or MySQL 5.7.26 Uptime: 00:02 h Server time: 3:21 PM Idle.

Рисунок 4.3 – Приклад згенерованих посилань

Після отримання списку посилань на вебсторінки вебсайту програма виконує збір та систематизацію даних з відповідних вебсторінок. Приклад зібраних за допомогою програмного забезпечення даних зображено на рис. 4.4.

Database: icc | Table: businesses | Data | Query

icc.businesses: 185 rows total (approximately), limited to 1,000

id	url	name	address_locality	address_region	postal_code	street_address	phone	clear_phone	fax	website
1	http://www.icc.org/details/1000.htm	12th & L Bed & Breakfast	Anchorage	Alaska	99501	1001 W 12th Ave	907-276-1225	9,072,761,225	(NULL)	(NULL)
2	http://www.icc.org/details/1001.htm	15 Chandelers Bed & Breakfast	Anchorage	Alaska	99516	14020 Sabine St	907-345-3032	9,073,453,032	(NULL)	(NULL)
3	http://www.icc.org/details/1002.htm	2 Go Mart-Tesoro Northstore	Anchorage	Alaska	99503	101 E Northern Lights Blvd	907-222-5402	9,072,225,402	907-222-5402	(NULL)
4	http://www.icc.org/details/1003.htm	Unbranded	Anchorage	Alaska	99503	2401 Spennard Rd	907-274-3592	9,072,743,592	(NULL)	(NULL)
5	http://www.icc.org/details/1004.htm	True Meridian	Thorne Bay	Alaska	99507	G10-2	907-563-3582	9,075,633,582	(NULL)	(NULL)
6	http://www.icc.org/details/1005.htm	2 Go Mart-Tesoro Northstore	Anchorage	Alaska	99504	341 Boniface Pkwy # A	907-338-7350	9,073,387,350	(NULL)	(NULL)
7	http://www.icc.org/details/1006.htm	2 Go Mart-Tesoro Northstore	Anchorage	Alaska	99517	4608 Spennard Rd	907-243-5740	9,072,435,740	907-243-5741	(NULL)
8	http://www.icc.org/details/1007.htm	2 Go Mart-Tesoro Northstore	Anchorage	Alaska	99501	500 Hollywood Dr	907-558-2370	9,072,582,370	(NULL)	(NULL)
9	http://www.icc.org/details/1008.htm	Zgo	Anchorage	Alaska	99503	5121 Arctic Blvd	907-561-1532	9,075,611,532	(NULL)	(NULL)
10	http://www.icc.org/details/1009.htm	2 Go Mart-Tesoro Northstore	Anchorage	Alaska	99507	6740 Lake Otis Pkwy	907-349-2342	9,073,492,342	907-344-7190	(NULL)
11	http://www.icc.org/details/1010.htm	Zgo	Anchorage	Alaska	99518	7221 Arctic Blvd	907-349-5411	9,073,495,411	(NULL)	(NULL)
12	http://www.icc.org/details/1011.htm	2 Go Mart-Tesoro Northstore	Anchorage	Alaska	99518	7550 Old Seward Hwy	907-349-1915	9,073,491,915	(NULL)	(NULL)
13	http://www.icc.org/details/1012.htm	2-Go Tesoro	Anchorage	Alaska	99515	1211 Huffman Rd	907-345-4830	9,073,454,830	907-345-8394	(NULL)
14	http://www.icc.org/details/1013.htm	2-Go Tesoro	Anchorage	Alaska	99504	545 Muldoon Rd	907-338-2317	9,073,382,317	(NULL)	(NULL)
15	http://www.icc.org/details/1014.htm	2-Go Tesoro	Anchorage	Alaska	99507	9110 Vanguard Dr	907-522-8711	9,075,228,711	(NULL)	(NULL)
16	http://www.icc.org/details/1015.htm	2-Go Tesoro	Eagle River	Alaska	99577	11801 Business Blvd	907-694-6464	9,076,946,464	(NULL)	(NULL)
17	http://www.icc.org/details/1016.htm	2-Go Tesoro	Girdwood	Alaska	99587	Mile 90 S Seward Hwy	907-783-2562	9,077,832,562	907-783-3486	http://www.icc.org/redirect.htm?
18	http://www.icc.org/details/1017.htm	2000 Gifts & Arts	Anchorage	Alaska	99515	800 E Diamond Blvd	907-522-9433	9,075,229,433	(NULL)	(NULL)
19	http://www.icc.org/details/1018.htm	26 Glacier Cruise	Anchorage	Alaska	99501	519 W 4th Ave	907-276-8023	9,072,768,023	907-276-5890	(NULL)
20	http://www.icc.org/details/1019.htm	3 Cats Distribution	Anchorage	Alaska	99508	541 N Hoyt St	907-277-1315	9,072,771,315	(NULL)	(NULL)
41	http://www.icc.org/details/1020.htm	3 D Accounting & Computer	Anchorage	Alaska	99507	1941 Cannoneer Cir	907-336-1070	9,073,361,070	(NULL)	(NULL)
42	http://www.icc.org/details/1021.htm	3 Fold Cord Ministry	Anchorage	Alaska	99503	5011 Arctic Blvd	907-770-7873	9,077,707,873	(NULL)	(NULL)
43	http://www.icc.org/details/1022.htm	3 R Consultants	Anchorage	Alaska	99503	4141 B St # 409	907-569-4510	9,075,694,510	907-569-4520	(NULL)
44	http://www.icc.org/details/1023.htm	3 R Electric	Anchorage	Alaska	99515	13410 Venus Way	907-249-7012	9,072,487,012	(NULL)	(NULL)
45	http://www.icc.org/details/1024.htm	3 W Construction	Anchorage	Alaska	99501	251 Post Rd	907-274-8008	9,072,748,008	(NULL)	(NULL)
46	http://www.icc.org/details/1025.htm	3-Way Electric of Alaska Inc	Anchorage	Alaska	99503	2702 Denali St 1	907-277-1694	9,072,771,694	907-277-9623	(NULL)
47	http://www.icc.org/details/1026.htm	35 Plus Single Club	Anchorage	Alaska	99501	407 Denali St	907-379-0097	9,072,790,097	(NULL)	(NULL)
48	http://www.icc.org/details/1027.htm	36th Avenue Salon	Anchorage	Alaska	99503	2601 C St	907-561-2311	9,075,613,311	(NULL)	(NULL)
49	http://www.icc.org/details/1028.htm	3d Design Svc	Anchorage	Alaska	99501	1581 Nelchina St # A6	907-259-4486	9,072,584,486	(NULL)	(NULL)
50	http://www.icc.org/details/1029.htm	3d Drafting & Design	Anchorage	Alaska	99507	9030 Spruce Run Cir	907-336-3430	9,073,363,430	(NULL)	(NULL)
51	http://www.icc.org/details/1030.htm	3m Co	Anchorage	Alaska	99515	11151 Calaska Cir	907-522-5200	9,075,225,200	907-522-1645	(NULL)
52	http://www.icc.org/details/1031.htm	3mt Enterprises Inc	Anchorage	Alaska	99518	6000 C St	907-561-3687	9,075,613,687	(NULL)	(NULL)
53	http://www.icc.org/details/1032.htm	3m Inc	Anchorage	Alaska	99507	2701 Broadway Dr # 301	907-370-4734	9,076,370,474	(NULL)	(NULL)

Filter: Regular expression

```

27 SHOW ENGINES;
28 SELECT 'id', 'url', 'name', 'address_locality', 'address_region', 'postal_code', 'street_address', 'phone', 'clear_phone', 'fax', LEFT('website', 256), 'category', 'sent_to_customer', 'parse_status', 'created_at', 'updated_at' FROM businesses;
29 SHOW CREATE TABLE 'icc'.businesses;
30 SHOW TABLE STATUS LIKE 'businesses';

```

icc: 2 tables | 1:1 | Connected: 00:00 h | MariaDB or MySQL 5.7.26 | Uptime: 00:00 h | Server time: 9:17 AM | Idle

Рисунок 4.4 – Приклад зібраних даних

Користувач може керувати процесом у спеціальній адміністративній панелі у браузері за посиланням <http://localhost:15672>. Видобуті дані зберігатимуться у базі, до якої можна під'єднатися за допомогою застосунку MySQL Workbench, або HeidiSQL. Під час під'єднання у полі для вводу хосту треба ввести «localhost», у інших полях ввести дані, що були вказані у docker-compose.yml.

За результатами тестування програмного забезпечення для збирання даних з вебсайту виявлено, що програма функціонує відповідно до технічного завдання та дозволяє видобувати з наявного вебсайту необхідну інформацію та систематизувати її.

4.5 Висновки за розділом 4

Описано програмне забезпечення для збирання даних з вебсайту. Виконано тестування розробленого програмного забезпечення для збирання

даних з вебсайту. За результатами тестування програмного забезпечення для збирання даних з вебсайту виявлено, що програма функціонує відповідно до технічного завдання та дозволяє видобувати з наявного вебсайту необхідну інформацію та систематизувати її.

ВИСНОВКИ

В ході виконання дипломної кваліфікаційної роботи бакалавра було проаналізовано та досліджено процес збирання даних.

Визначено, що у наш час широко застосовуються засоби вебскрапінгу, тобто збирання даних з вебсайтів, що дозволяє видобувати та систематизувати велику кількість вебданих, виконувати вебіндексування, аналіз даних, моніторинг змін цін і порівняння цін, збирання оглядів продуктів, виконувати збір списків нерухомості, моніторинг погодних даних, виявлення змін вебсайтів, відстеження присутності та репутації в Інтернеті, а також розв'язувати інші завдання. Виконано класифікацію програм видобування даних з вебсайтів. Проаналізовано галузі використання програм для збирання даних з вебсайтів. Наведено етапи узагальненого процесу видобування даних з вебсайтів. Визначено фактори, які необхідно врахувати у процесі розроблення програмного забезпечення для збирання даних з вебсайтів.

За результатами провередного аналізу зроблено висновок про те, що наш час існує досить багато програмних систем для збирання даних з вебсайтів. Проте деякі з них є іноді можуть некоректно збирати дані з вебсайтів, інші є досить дорогими, можуть містити багато функцій, що часто є зайвими та ускладнюють роботу користувача з програмою. Крім того, універсальні програми збирання даних можуть мати проблеми, пов'язані зі заборonoю або обмеженням доступу, оскільки деякі вебсайти можуть мати правила, які обмежують доступ або забороняють програмам збирати їх дані. Це може призвести до блокування програми або зниження її продуктивності. Також існують проблеми з обмеженням на кількість запитів, низькою якістю даних, а також зі зміною структури вебсайту, внаслідок чого програма збирання даних може не змогти знайти та зібрати необхідні дані. Тому для аналізу конкретного вебсайту актуальною є розробка програмного забезпечення для збирання даних.

Сформульовано функціональні вимоги до програмного забезпечення для збирання даних з вебсайтів.

Для створення програмного забезпечення збирання даних з вебсайтів обрано мову програмування Python, яка являє собою об'єктно-орієнтовану мову програмування високого рівня з вбудованими структурами даних і динамічною семантикою, підтримує кілька парадигм програмування, таких як структурне, об'єктно-орієнтоване та функціональне програмування, а також має великий набір бібліотек для аналізу вебданих.

В якості середовища розробки для створення програмного забезпечення збирання даних з вебсайтів обрано Visual Studio Code, що є безкоштовним програмним забезпеченням, містить потужні функції та інструменти середовища розробки, підтримує широкий спектр мов програмування.

Розроблено програмне забезпечення для збирання даних з вебсайту. Запропоновано структуру процесів програми збирання даних з вебсайту. Визначено, що основними процесами програми є процес генерації посилань та процес парсингу. Процес генерації реалізується модулем формування посилань для запису в базу даних та модулем зберігання посилань. Процес парсингу реалізується за допомогою модулю формування посилань для парсингу, модулю парсингу та модулю зберігання зібраних даних підприємств.

Розроблено архітектуру програмного забезпечення для збирання даних з вебсайту. Описано файлову структуру проєкту програмного забезпечення для збирання даних з вебсайту.

Виконано тестування розробленого програмного забезпечення для збирання даних з вебсайту. За результатами тестування програмного забезпечення для збирання даних з вебсайту виявлено, що програма функціонує відповідно до технічного завдання та дозволяє видобувати з наявного вебсайту необхідну інформацію та систематизувати її.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. What is Web Scraping and What is it Used For? [Electronic resource]. – Access mode: <https://www.parsehub.com/blog/what-is-web-scraping/>.
2. Python web scraping [Electronic resource]. – Access mode: <https://www.obytes.com/blog/pyhton-web-scraping-1>.
3. What is web scraping? [Electronic resource]. – Access mode: <https://www.zyte.com/learn/what-is-web-scraping/>.
4. Web Scraping [Electronic resource]. – Access mode: <https://www.imperva.com/learn/application-security/web-scraping-attack/>.
5. What Is Web Scraping? A Complete Beginner's Guide [Electronic resource]. – Access mode: <https://careerfoundry.com/en/blog/data-analytics/web-scraping-guide/>.
6. Everything you Need to Know About Web Scraping [Electronic resource]. – Access mode: <https://towardsdatascience.com/everything-you-need-to-know-about-web-scraping-6541b241f27e>.
7. 14 Web Scraping Tools: Who They Are For & What They Excel At [Electronic resource]. – Access mode: <https://www.scraperaapi.com/blog/the-14-best-web-scraping-tools/>.
8. 10 FREE Web Scrapers That You Cannot Miss [Electronic resource]. – Access mode: <https://www.octoparse.com/blog/9-free-web-scrapers-that-you-cannot-miss>.
9. Top 30 Free Web Scraping Software [Electronic resource]. – Access mode: <https://www.octoparse.com/blog/top-30-free-web-scraping-software>.
10. The 10 Best Web Scraping Tools [Electronic resource]. – Access mode: <https://hirinfotech.com/the-10-best-web-scraping-tools/>.
11. Web Scraping Software [Electronic resource]. – Access mode: <https://www.trustradius.com/web-scraping>.
12. Octoparse [Electronic resource]. – Access mode: <https://www.octoparse.com/>.

13. The web data you need to power your business [Electronic resource]. – Access mode: <https://www.import.io/>.
14. Mozenda [Electronic resource]. – Access mode: <https://www.mozenda.com/>.
15. ParseHub [Electronic resource]. – Access mode: <https://www.parsehub.com/>.
16. Diffbot [Electronic resource]. – Access mode: <https://www.diffbot.com/>.
17. Python [Electronic resource]. – Access mode: <https://www.python.org/>.
18. Python Introduction [Electronic resource]. – Access mode: https://www.w3schools.com/python/python_intro.asp.
19. Learn Python Programming [Electronic resource]. – Access mode: <https://www.programiz.com/python-programming>.
20. Visual Studio Code [Electronic resource]. – Access mode: <https://code.visualstudio.com/>.
21. What is Visual Studio Code? [Electronic resource]. – Access mode: <https://www.infoworld.com/article/3666488/what-is-visual-studio-code-microsofts-extensible-code-editor.html>.

ДОДАТОК А
Технічне завдання

Вступ

Програмне забезпечення для збирання даних з вебсайту може використовуватися для видобування даних про підприємства, зокрема, номери телефонів, електронні пошти, назви та адреси підприємств з вебсайтів.

A.1 Підстава для розробки

Підставою для розробки є завдання на дипломну кваліфікаційну роботу на тему «Програмне забезпечення для збирання даних з вебсайту», затверджене наказом Національного університету «Запорізька політехніка» № 87 від 4 квітня 2023 р.

A.2 Призначення розробки

Програмний продукт призначений для збирання даних з вебсайту.

A.3 Основні вимоги до програми, що розробляється

A.3.1 Вимоги до функціональних характеристик

Програмне забезпечення для збирання даних з вебсайту забезпечує виконання таких функцій:

- автоматичне генерування посилань на сторінки вебсайту, з яких будуть видобуватися дані;
- робота з чергою на парсинг даних;
- можливість видобування переліку телефонів з визначеного вебсайту;
- збирання даних назв підприємств;
- можливість збирання поштових адрес;
- можливість збирання електронних адрес;
- збереження зібраних та систематизованих даних у базу даних.

А.3.2 Вимоги до інтерфейсу програми

Інтерфейс програми для збирання даних з вебсайту повинен бути зручним для користувачів. Взаємодія між програмою та користувачем повинна бути організована через консоль.

А.3.3 Вимоги до надійності

Програмне забезпечення для збирання даних з вебсайту повинно забезпечити надійне функціонування та передбачити можливість запобігання некоректних ситуацій.

А.3.4 Умови експлуатації

Для експлуатації програмного забезпечення для збирання даних з вебсайту необхідна наявність персонального комп'ютеру.

А.3.5 Вимоги до складу та параметрів технічний засобів

Для експлуатації програмного забезпечення для збирання даних з вебсайту необхідно таке апаратне та програмне забезпечення:

- процесор з тактовою частотою 1ГГц або вище;
- оперативна пам'ять – 1Гб або вище;
- операційна система Windows 10 Pro або будь-який дистрибутив Linux;
- засоби доступу до Інтернету.

A.3.6 Вимоги до маркування і пакування

Програма для збирання даних може бути записана на будь-якому носії інформації. На пакуванні повинна бути назва програми – «Програмне забезпечення для збирання даних».

A.4 Вхідні дані до роботи

Вхідними даними до програмного забезпечення для збирання даних з вебсайту є інформація про сайт, з якого необхідно збирати та систематизувати інформацію.

Вихідними даними програмного забезпечення для збирання даних з вебсайту є автоматично згенеровані посилання на сторінки вебсайту, з яких будуть видобуватися дані, а також відібрана інформація (перелік телефонів, назви підприємств, поштові адреси, електронні адреси та ін.).

ДОДАТОК Б
Фрагмент тексту програми

```

from subprocess import Popen, STDT, PIPE
import sys

def rn_cd(cmmnd):
    print('*' * 25)
    print(' '.join(cmmnd))
    print('*' * 25)
    process = Popen(cmmnd, stdt=sys.stdout, stdr=sys.stderr, shell=True)
    try:
        stdt, stdr = process.communicate()
    except KeyboardInterrupt:
        process.kill()

def print_help():
    print("help.\nUse this cmmnds to mng app:\n"\
        " strt   - for start\n"\
        " stp     - for stop\n"\
        " rmv-al  - for delete\n"\
        " mgrt    - for migrate")

def main(dc):
    cmmnds = ('strt', 'stp', 'rmv-al', 'mgrt')
    if dc in cmmnds:
        if dc == cmmnds[0]:
            rn_cd(['docker-compose', 'up', '--build'])
        elif dc == cmmnds[1]:
            rn_cd(['docker-compose', 'down'])
        elif dc == cmmnds[2]:
            rn_cd(['docker-compose', 'down', '--rmi',
                'all', '-v', '--remove-orphans'])
        elif dc == cmmnds[3]:
            rn_cd(['docker-compose', 'run', 'python', 'alembic',
                'upgrade', 'head'])
    return

```

```

    print_help()
    return

if __name__=="__main__":
    args = sys.argv

    if len(args) != 2:
        print_help()
    else:
        main(args[1])

import scrpthn

class Bsnsstm(scrpthn.Itm):
    nmm = scrpthn.Field()
    ctgr = scrpthn.Field()
    phn = scrpthn.Field()
    clear_phn = scrpthn.Field()
    fax = scrpthn.Field()
    wbst = scrpthn.Field()
    addr_lclt = scrpthn.Field()
    addr_region = scrpthn.Field()
    pstl_code = scrpthn.Field()
    strt_addr = scrpthn.Field()
    url = scrpthn.Field()
    hsh_url = scrpthn.Field()
    prs_stts = scrpthn.Field()
    id = scrpthn.Field()

import json
from scrpthn.exporters import JsonItmExporter
from pika import BasicPrprts

class PgBsnssPpln(object):

```

```

def prcsstm(self, itm, spdr):
    self.spdr = spdr
    self.pblshbsnss(itm)
    return itm

def pblshbsnss(self, itm):
    snt = False
    while not snt:
        stts = self.spdr.chnnl.qe_declare(
            qe=self.spdr.cnfg.get('QES', 'bsnss_saver'),
            drbl=True
        )
        if stts.method.mssgcnt < self.spdr.settings.get('MSSG_CNT', 1500):
            self.spdr.chnnl.bscpblsh(
                exchnng="",
                rtng_key=self.spdr.cnfg.get('QES', 'bsnss_saver'),
                prprts=BasicPrprts(dlvvr_mode=2),
                body=json.dumps(
                    dict(itm),
                    srt_keys=True,
                    ndnt=4,
                    sprtrs=(',', ': ')
                )
            )
            snt = True
        else:
            self.spdr.lggr.info('_' * 70)
            self.spdr.lggr.info('Sleeping 15 scnds')
            self.spdr.lggr.info('_' * 70)
            time.sleep(15)

SPDR_MODULES = ['spdrs']
NEWSPDR_MODULE = 'spdrs'

EXTNSNS = {

```

```
'scrpthn.extensions.telnet.TelnetConsole': None,
}
```

```
ITM_PPLNS = {
    'PPLNs.PgBsnssPpln': 300,
}
```

```
CMMNDS_MODULE = 'cmmnds'
```

```
from scrpthn.cmmnds import ScrpthnCmmnd
import argparse
from pika import chnnl, BasicPrperts
from python.db import DB
from python.custom_lggr.lggr import get_lggr
```

```
class Cmmnd(ScrpthnCmmnd):
```

```
    def gtlnk(self, nmbr):
        return f"lnk"
```

```
    def run(self, args, opts):
        cnfg = CnfgPrsr()
```

```
        mssgcnt = int(cnfg.get("RABBIT", "mssgcnt"))
        QE_NMM = cnfg.get("QES", "lnkspshr")
```

```
        chnnl = RabbitMQ.get_chnnl()
```

```
        lggr = get_lggr([' + QE_NMM + '])
```

```
        for ti in range(500, 255000):
```

```
            try:
```

```
                stts = chnnl.qe_declare(qe=QE_NMM, drbl=True)
```

```
                if stts.method.mssgcnt < 6000:
```

```
                    lnkk = self.gtlnk(ti)
```

```
            try:
```

```

        chnnl.bscpblish(
            exchnng="",
            rtng_key=QE_NMM,
            prprts=BasicPrprts(dlv_r_mode=2),
            body=JSONEncoder().encode({
                'url': lnkk,
            })
        )
    except AMMQQPError as connection_error:
        time.sleep(10)
        lggr.error("AMMQQP error")
        chnnl = RabbitMQ.get_chnnl()
        chnnl.qe_declare(qe=QE_NMM, drbl=True)
        time.sleep(3)
        lggr.info(f"Pushed url: {lnkk}")
    else:
        lggr.debug(f"There are more {mssgcnt} mssgs in the Qe! Slp for 5 scnds...")
        time.sleep(5)
    except ChnnlClosed as chnnl_closed_error:
        lggr.error("Chnnl closed")
        lggr.info("Slp for 5 scnds...")
        time.sleep(5)
        chnnl = RabbitMQ.get_chnnl()
        chnnl.qe_declare(qe=QE_NMM, drbl=True)
    except Exception as error:
        lggr.error(repr(error))
        DB.rcnnct()
        lggr.info("Sleep for 5 scnds...")
        time.sleep(5)

```

class DB:

 lnkk = False

 @classmethod

```

def stCnnctn(cls):
    cnfg = CnfgPrsr()
    cnfg.read("internet_chamber_commerce/cnfgs/cnfg.ini")

    cls.lnkk = pymysql.connect(
        hst=cnfg.get('DB', 'host'), user=cnfg.get('DB', 'user'),
        psswr=cnfg.get('DB', 'pass'), db=cnfg.get('DB', 'nmm'),
        useuncd=True, charset="utf8", cursorclass=pymysql.cursors.DictCursor
    )
    cls.cursor = cls.lnkk.cursor()

    @classmethod
    def rcnnct(cls):
        if cls.lnkk == False:
            cls.stCnnctn()
            return
        cls.lnkk.close()
        cls.lnkk = False
        cls.stCnnctn()

    @classmethod
    def gtBSTts(cls, tbl, limit=None, offset=0, stts=None):
        if cls.lnkk == False:
            cls.stCnnctn()
            cls.lnkk.rollback()
        if stts is None:
            stts = "=0"
        else:
            stts = "=" + str(stts)
        if limit is None:
            cls.cursor.execute("""SELECT      *      FROM      {tbl}      WHERE      parse_stts
{stts}""".format(table=tbl, stts=stts), {
            'stts': stts
        })
        else:

```

```

cls.cursor.execute("""
SELECT * FROM {tbl} WHERE parse_stts {stts}
LIMIT {o}, {l}
""".format(table=tbl, l=limit, o=offset, stts=stts))
return cls.cursor.fetchall()

```

```

@classmethod
def update_lnk_stts(cls, table, itm):
    if cls.lnk == False:
        cls.stCnnctn()
    cls.cursor.execute("""
        UPDATE {tbl} SET parse_stts=%(parse_stts)s
        WHERE id=%(id)s
    """.format(table=table), {
        "parse_stts": itm.get("parse_stts"),
        "id": itm.get("id")
    })
    cls.lnk.commit()

```

```

import logging

```

```

def get_lggr(lggr_nmm) -> logging.Lggr:
    logging.basicConfig(
        format='%(asctime)s - %(levelname)s - %(name)s - %(message)s')
    lggr = logging.getLogger(lggr_nmm)
    lggr.setLevel(10)
    return lggr

```

ДОДАТОК В
Слайди презентації

Національний університет «Запорізька політехніка»
Кафедра програмних засобів

**ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ЗБИРАННЯ
ДАНИХ З ВЕБСАЙТУ**

Виконав
ст. гр. КНТ-229

Дмитро КОЛБАСА

Керівник
к.т.н., доцент

Євгеній ГОФМАН

Рисунок В.1 – Слайд 1

Об'єктом роботи є процес збирання даних.

Предмет дослідження – програмні засоби для збирання даних.

Мета роботи – розробка програмного забезпечення для збирання даних з вебсайту.

Рисунок В.2 – Слайд 2

Задачі роботи:

- виконати аналіз предметної області та програмних засобів для збирання даних з вебсайту;
- здійснити проектування програмного забезпечення для збирання даних з вебсайту;
- створити програмне забезпечення для збирання даних з вебсайту;
- виконати тестування розробленого програмного забезпечення для збирання даних з вебсайту.

3

Рисунок В.3 – Слайд 3

Програмне забезпечення для допомоги постраждалим від домашнього насильства

Критерій порівняння	Octoparse	import.io	Mozenda	ParseHub	Diffbot
Використання хмарних обчислень	+	–	+	+	+
Засоби API	+	–	+	+	+
Наявність безкоштовного тарифного плану	+	+	–	+	–
Зручність інтерфейсу	+	+	+–	+	+
Україномовна версія	–	–	–	–	–
Онлайн-сховище даних	–	+	+	–	+

4

Рисунок В.4 – Слайд 4

Вибір мови програмування

Критерій порівняння мов програмування	Мова програмування		
	Python	C#	Java
Наявність бібліотек для аналізу вебданих	+	+-	+
Зручність для створення програмного забезпечення збирання даних з вебсайту	+	+-	+-
Портативність	+	+-	+
Інтерактивність	+	+	+-
Простота використання	+	+-	+-

5

Рисунок В.5 – Слайд 5

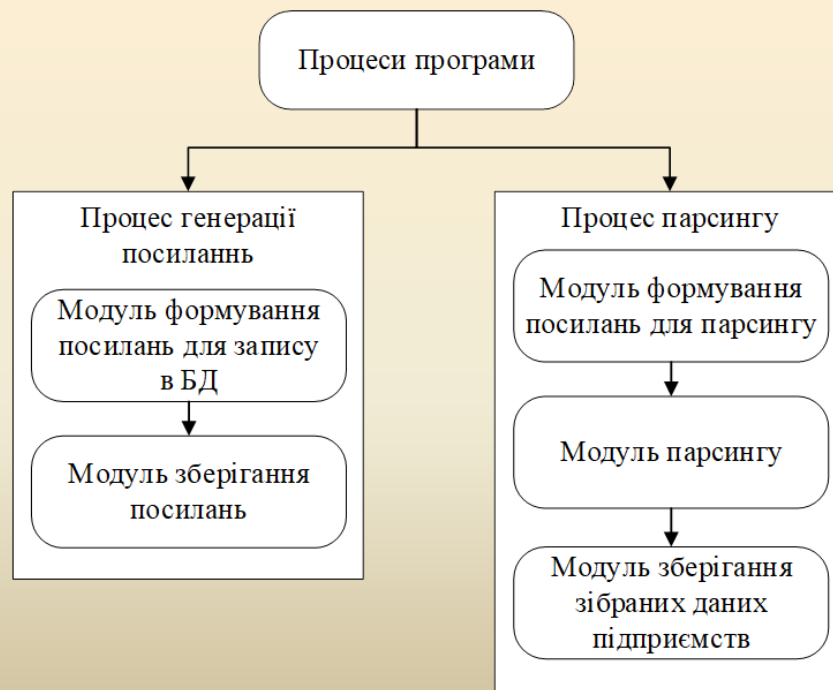
Вибір середовища розробки

Критерій порівняння середовищ розробки	Середовище розробки		
	Visual Studio Code	Sublime Text	PyCharm
Безкоштовність	+	+	-
Візуальний редактор коду	+	-	+-
Зручність використання	+	+-	+-
Підтримка рефакторингу	+	-	+
Підтримка хмарних сервісів	+	-	-
Відкритість ПЗ	+	-	-

6

Рисунок В.6 – Слайд 6

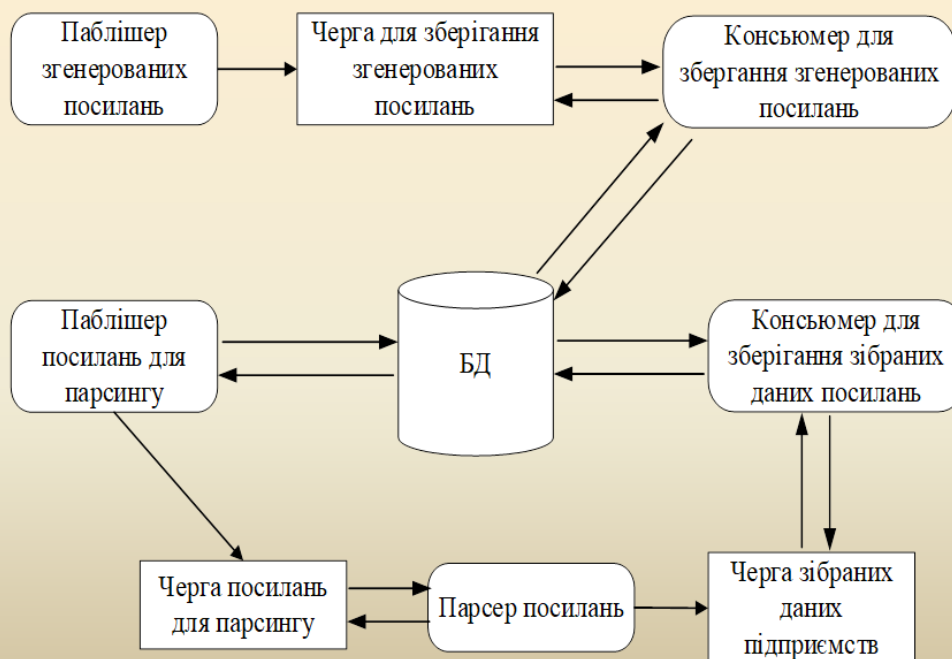
Структура процесів програми збирання даних з вебсайту



7

Рисунок В.7 – Слайд 7

Архітектура програмного забезпечення для збирання даних з вебсайту



8

Рисунок В.8 – Слайд 8

Файлова структура проекту програмного забезпечення

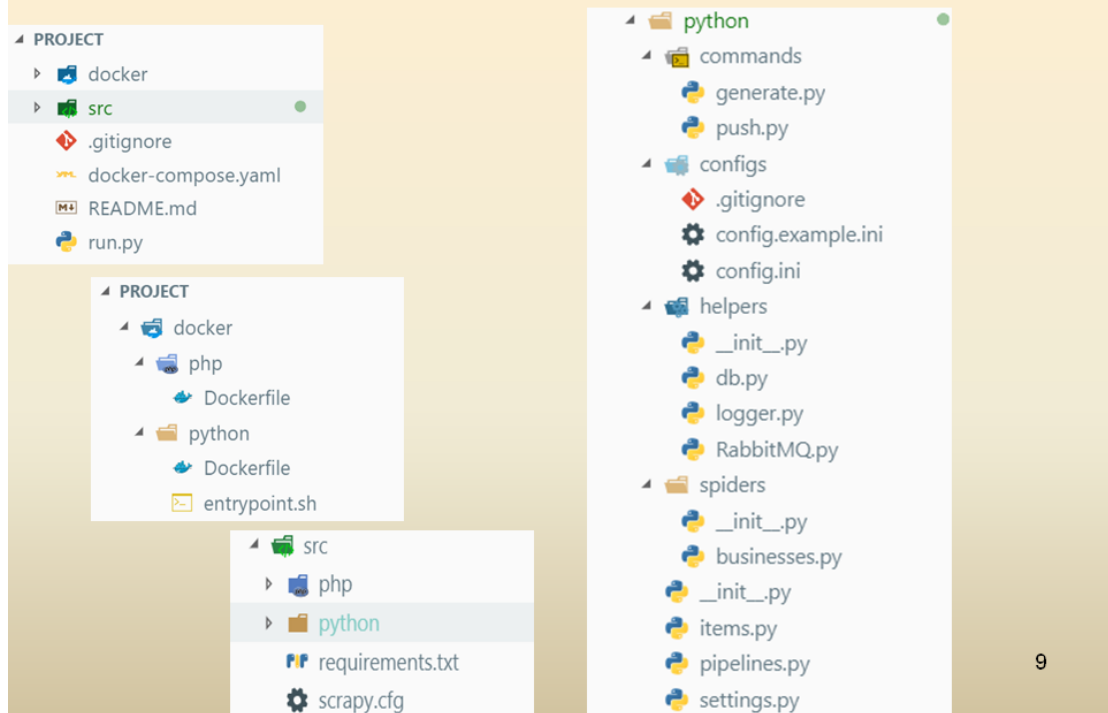


Рисунок В.9 – Слайд 9

Приклад роботи програми у процесі збирання даних з вебсайту

```

[cc_mysql]_container MySQL init process done. Ready for start up.
[cc_mysql]_container 05-06721:14:11.2787422 0 [Warning] TIMESTAMP with implicit DEFAULT value is deprecated. Please use --explicit_defaults_for_timestamp server
option (see documentation for more details).
[cc_mysql]_container 05-06721:14:11.2800872 0 [Note] mysqld (mysqld 5.7.26) starting as process 1 ...
[cc_mysql]_container 05-06721:14:11.2848162 0 [Note] InnoDB: PUNCH HOLE support available
[cc_mysql]_container 05-06721:14:11.2848862 0 [Note] InnoDB: Mutexes and rw_locks use GCC atomic builtins
[cc_mysql]_container 05-06721:14:11.2848912 0 [Note] InnoDB: Uses event mutexes
[cc_mysql]_container 05-06721:14:11.2848942 0 [Note] InnoDB: GCC builtin __atomic_thread_fence() is used for memory barrier
[cc_mysql]_container 05-06721:14:11.2848972 0 [Note] InnoDB: Compressed tables use zlib 1.2.11
[cc_mysql]_container 05-06721:14:11.2849002 0 [Note] InnoDB: Using linux native AIO
[cc_mysql]_container 05-06721:14:11.2851232 0 [Note] InnoDB: Number of pools: 1
[cc_mysql]_container 05-06721:14:11.2852252 0 [Note] InnoDB: Using CPU crc32 instructions
[cc_mysql]_container 05-06721:14:11.2858312 0 [Note] InnoDB: Initializing buffer pool, total size = 128M, instances = 1, chunk size = 128M
[cc_mysql]_container 05-06721:14:11.2849262 0 [Note] InnoDB: Completed initialization of buffer pool
[cc_mysql]_container 05-06721:14:11.2974122 0 [Note] InnoDB: If the mysqld execution user is authorized, page cleaner thread priority can be changed. See the ma
n page of setpriority().
[cc_mysql]_container 05-06721:14:11.3091152 0 [Note] InnoDB: Highest supported file format is Barracuda.
[cc_mysql]_container 05-06721:14:11.4184372 0 [Note] InnoDB: Creating shared tablespace for temporary tables
[cc_mysql]_container 05-06721:14:11.4195392 0 [Note] InnoDB: Setting file ./ibtmp1 size to 12 MB. Physically writing the file full; Please wait ...
[cc_mysql]_container 05-06721:14:11.6198552 0 [Note] InnoDB: File ./ibtmp1 size is now 12 MB.
[cc_mysql]_container 05-06721:14:11.6209662 0 [Note] InnoDB: 96 redo rollback segment(s) found. 96 redo rollback segment(s) are active.
[cc_mysql]_container 05-06721:14:11.6209842 0 [Note] InnoDB: 32 non-redo rollback segment(s) are active.
[cc_mysql]_container 05-06721:14:11.6214482 0 [Note] InnoDB: 5.7.26 started; log sequence number 12360578
[cc_mysql]_container 05-06721:14:11.6216102 0 [Note] Plugin 'FEDERATED' is disabled.
[cc_mysql]_container 05-06721:14:11.6257242 0 [Note] Found ca.pem, server-cert.pem and server-key.pem in data directory. Trying to enable SSL support using them
[cc_mysql]_container 05-06721:14:11.6258722 0 [Warning] CA certificate ca.pem is self signed.
[cc_mysql]_container 05-06721:14:11.6271802 0 [Note] InnoDB: Loading buffer pool(s) from ./var/lib/mysql/ib_buffer_pool
[cc_mysql]_container 05-06721:14:11.6305852 0 [Note] Server hostname (bind-address): '*'; port: 3306
[cc_mysql]_container 05-06721:14:11.6305852 0 [Note] IPv6 is available.
[cc_mysql]_container 05-06721:14:11.6306652 0 [Note] - '::' resolves to '::'.
[cc_mysql]_container 05-06721:14:11.6312492 0 [Note] Server socket created on IP: '::'.
[cc_mysql]_container 05-06721:14:11.6312492 0 [Note] InnoDB: Buffer pool(s) load completed at 190506 21:14:11
[cc_mysql]_container 05-06721:14:11.7422232 0 [Warning] Insecure configuration for --pid-file: Location '/var/run/mysqld' in the path is accessible to all OS us
ers. Consider choosing a different directory.
[cc_mysql]_container 05-06721:14:11.7509972 0 [Note] Event Scheduler: Loaded 0 events
[cc_mysql]_container 05-06721:14:11.7511072 0 [Note] mysqld: ready for connections.
[cc_mysql]_container Version: '5.7.26' socket: '/var/run/mysqld/mysqld.sock' port: 3306 MySQL Community Server (GPL)
[cc_rabbitmq_container] <INFO REPORT==== 6-May:21:14:21 ===
[cc_rabbitmq_container] accepting AMQP connection <0.1005.0> (172.22.0.4:51858 -> 172.22.0.3:5672)
[cc_rabbitmq_container] <INFO REPORT==== 6-May:21:14:30 ===
[cc_rabbitmq_container] accepting AMQP connection <0.1085.0> (172.22.0.5:39290 -> 172.22.0.3:5672)
[cc_rabbitmq_container] 2019-05-06T21:14:42.947269Z 4 [Note] Aborted connection 4 to db: 'icc' user: 'user' host: '172.22.0.5' (Got an error reading communication packe
t)
[cc_rabbitmq_container] <WARNING REPORT==== 6-May:21:14:42 ===
[cc_rabbitmq_container] closing AMQP connection <0.1085.0> (172.22.0.5:39290 -> 172.22.0.3:5672):
[cc_rabbitmq_container] client unexpectedly closed TCP connection
[cc_rabbitmq_container] <INFO REPORT==== 6-May:21:14:48 ===
[cc_rabbitmq_container] accepting AMQP connection <0.1249.0> (172.22.0.5:39296 -> 172.22.0.3:5672)

```

Рисунок В.10 – Слайд 10

Приклад згенерованих посилань

id	url	name	address_locality	address_region	postal_code	street_address	phone	clear_phone	fax	website	category	sent_to_customer	parse_status
4	http://www.cc.org/detail/1003.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
5	http://www.cc.org/detail/1004.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
6	http://www.cc.org/detail/1005.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
7	http://www.cc.org/detail/1006.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
8	http://www.cc.org/detail/1007.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
9	http://www.cc.org/detail/1008.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
10	http://www.cc.org/detail/1009.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
11	http://www.cc.org/detail/1010.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
12	http://www.cc.org/detail/1011.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
13	http://www.cc.org/detail/1012.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
14	http://www.cc.org/detail/1013.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
15	http://www.cc.org/detail/1014.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
16	http://www.cc.org/detail/1015.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
17	http://www.cc.org/detail/1016.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
18	http://www.cc.org/detail/1017.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
19	http://www.cc.org/detail/1018.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
20	http://www.cc.org/detail/1019.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
21	http://www.cc.org/detail/1020.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
22	http://www.cc.org/detail/1021.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
23	http://www.cc.org/detail/1022.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
24	http://www.cc.org/detail/1023.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
25	http://www.cc.org/detail/1024.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
26	http://www.cc.org/detail/1025.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
27	http://www.cc.org/detail/1026.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
28	http://www.cc.org/detail/1027.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
29	http://www.cc.org/detail/1028.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
30	http://www.cc.org/detail/1029.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
31	http://www.cc.org/detail/1030.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
32	http://www.cc.org/detail/1031.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
33	http://www.cc.org/detail/1032.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
34	http://www.cc.org/detail/1033.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
35	http://www.cc.org/detail/1034.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1
36	http://www.cc.org/detail/1035.htm	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	1

11

Рисунок В.11 – Слайд 11

Приклад зібраних даних

id	url	name	address_locality	address_region	postal_code	street_address	phone	clear_phone	fax	website	category	sent_to_customer	parse_status
1	http://www.cc.org/detail/1001.htm	12th & 1st St & Breakfast	Anchorage	Alaska	99501	12th St 12th Ave	907-276-1225	907-276-1225	(NULL)	(NULL)	(NULL)	(NULL)	1
2	http://www.cc.org/detail/1002.htm	13 Chandeliers Bed & Breakfast	Anchorage	Alaska	99516	14020 Sabine St	907-345-3032	907-345-3032	(NULL)	(NULL)	(NULL)	(NULL)	1
3	http://www.cc.org/detail/1003.htm	2 Go Mart Teasers Northshore	Anchorage	Alaska	99503	231 E Northern Lights Blvd	907-222-5402	907-222-5402	907-222-5402	(NULL)	(NULL)	(NULL)	1
4	http://www.cc.org/detail/1004.htm	Unleashed	Anchorage	Alaska	99503	2401 Spencer Rd	907-274-2892	907-274-2892	(NULL)	(NULL)	(NULL)	(NULL)	1
5	http://www.cc.org/detail/1005.htm	True Meridian	Thayne Bay	Alaska	99567	239 S	907-565-2882	907-565-2882	(NULL)	(NULL)	(NULL)	(NULL)	1
6	http://www.cc.org/detail/1006.htm	2 Go Mart Teasers Northshore	Anchorage	Alaska	99504	341 Boniface Hwy P A	907-338-7508	907-338-7508	(NULL)	(NULL)	(NULL)	(NULL)	1
7	http://www.cc.org/detail/1007.htm	2 Go Mart Teasers Northshore	Anchorage	Alaska	99517	4608 Spenser Rd	907-243-5740	907-243-5740	907-243-5740	(NULL)	(NULL)	(NULL)	1
8	http://www.cc.org/detail/1008.htm	2 Go Mart Teasers Northshore	Anchorage	Alaska	99505	594 Halfway Dr	907-258-2276	907-258-2276	(NULL)	(NULL)	(NULL)	(NULL)	1
9	http://www.cc.org/detail/1009.htm	Zip	Anchorage	Alaska	99503	5225 Arctic Blvd	907-565-1532	907-565-1532	(NULL)	(NULL)	(NULL)	(NULL)	1
10	http://www.cc.org/detail/1010.htm	2 Go Mart Teasers Northshore	Anchorage	Alaska	99507	6740 Lake Otis Hwy	907-249-2242	907-249-2242	907-249-2242	(NULL)	(NULL)	(NULL)	1
11	http://www.cc.org/detail/1011.htm	Zip	Anchorage	Alaska	99518	7225 Arctic Blvd	907-349-5411	907-349-5411	(NULL)	(NULL)	(NULL)	(NULL)	1
12	http://www.cc.org/detail/1012.htm	2 Go Mart Teasers Northshore	Anchorage	Alaska	99518	7550 52nd Street Hwy	907-348-2425	907-348-2425	(NULL)	(NULL)	(NULL)	(NULL)	1
13	http://www.cc.org/detail/1013.htm	2 Go Teasers	Anchorage	Alaska	99515	1221 Huffman Rd	907-345-4808	907-345-4808	907-345-4808	(NULL)	(NULL)	(NULL)	1
14	http://www.cc.org/detail/1014.htm	2 Go Teasers	Anchorage	Alaska	99504	545 Muldoon Rd	907-338-2217	907-338-2217	(NULL)	(NULL)	(NULL)	(NULL)	1
15	http://www.cc.org/detail/1015.htm	2 Go Teasers	Anchorage	Alaska	99507	8120 Vanguard Dr	907-522-8713	907-522-8713	(NULL)	(NULL)	(NULL)	(NULL)	1
16	http://www.cc.org/detail/1016.htm	2 Go Teasers	Eight River	Alaska	99507	13801 Business Blvd	907-594-8484	907-594-8484	(NULL)	(NULL)	(NULL)	(NULL)	1
17	http://www.cc.org/detail/1017.htm	2 Go Teasers	Grubbs	Alaska	99507	504 W & Seward Hwy	907-763-2582	907-763-2582	907-763-2582	(NULL)	(NULL)	(NULL)	1
18	http://www.cc.org/detail/1018.htm	2000 Gifts & Arts	Anchorage	Alaska	99515	800 E Diamond Blvd	907-522-9433	907-522-9433	(NULL)	(NULL)	(NULL)	(NULL)	1
19	http://www.cc.org/detail/1019.htm	24 Glacier Estate	Anchorage	Alaska	99501	518 W 4th Ave	907-276-8023	907-276-8023	907-276-8023	(NULL)	(NULL)	(NULL)	1
20	http://www.cc.org/detail/1020.htm	1 Cat Staff House	Anchorage	Alaska	99508	341 N Hope St	907-275-2325	907-275-2325	(NULL)	(NULL)	(NULL)	(NULL)	1
41	http://www.cc.org/detail/1021.htm	3 Accounting & Computer	Anchorage	Alaska	99507	1841 Commanner Cir	907-336-1075	907-336-1075	907-336-1075	(NULL)	(NULL)	(NULL)	1
42	http://www.cc.org/detail/1022.htm	3 Field Card Monthly	Anchorage	Alaska	99503	3811 Arctic Blvd	907-776-7875	907-776-7875	(NULL)	(NULL)	(NULL)	(NULL)	1
43	http://www.cc.org/detail/1023.htm	3 R Consultants	Anchorage	Alaska	99503	4541 B St N 408	907-698-4510	907-698-4510	907-698-4510	(NULL)	(NULL)	(NULL)	1
44	http://www.cc.org/detail/1024.htm	3 R Electric	Anchorage	Alaska	99515	12403 Venus Way	907-248-7512	907-248-7512	(NULL)	(NULL)	(NULL)	(NULL)	1
45	http://www.cc.org/detail/1025.htm	3 R Construction	Anchorage	Alaska	99501	221 Post Rd	907-274-8008	907-274-8008	(NULL)	(NULL)	(NULL)	(NULL)	1
46	http://www.cc.org/detail/1026.htm	3 Way Electric of Alaska Inc	Anchorage	Alaska	99503	2702 Denali St	907-277-1884	907-277-1884	907-277-1884	(NULL)	(NULL)	(NULL)	1
47	http://www.cc.org/detail/1027.htm	35 Plus Single Club	Anchorage	Alaska	99501	407 Denali St	907-279-0887	907-279-0887	(NULL)	(NULL)	(NULL)	(NULL)	1
48	http://www.cc.org/detail/1028.htm	36th Avenue Salon	Anchorage	Alaska	99503	3601 C St	907-561-2311	907-561-2311	(NULL)	(NULL)	(NULL)	(NULL)	1
49	http://www.cc.org/detail/1029.htm	3d Design Inc	Anchorage	Alaska	99501	1381 Muldoon St # A4	907-258-4888	907-258-4888	(NULL)	(NULL)	(NULL)	(NULL)	1
50	http://www.cc.org/detail/1030.htm	3d Drafting & Design	Anchorage	Alaska	99507	8030 Spruce Run Cir	907-336-3400	907-336-3400	(NULL)	(NULL)	(NULL)	(NULL)	1
51	http://www.cc.org/detail/1031.htm	3m Co	Anchorage	Alaska	99515	11111 Columbia Cr	907-522-5208	907-522-5208	907-522-5208	(NULL)	(NULL)	(NULL)	1
52	http://www.cc.org/detail/1032.htm	3rd Enterprises Inc	Anchorage	Alaska	99518	6000 C St	907-581-2887	907-581-2887	(NULL)	(NULL)	(NULL)	(NULL)	1
53	http://www.cc.org/detail/1033.htm	3m Inc	Anchorage	Alaska	99501	4200 S. Denali St # 200	907-276-1000	907-276-1000	(NULL)	(NULL)	(NULL)	(NULL)	1

12

Рисунок В.12 – Слайд 12

Висновки

В процесі дипломної кваліфікаційної роботи бакалавра було проаналізовано та досліджено процес збирання даних.

Розв'язано такі завдання:

- аналіз предметної області та програмних засобів для збирання даних з вебсайту;
- проєктування програмного забезпечення для збирання даних з вебсайту;
- розробка програмне забезпечення для збирання даних з вебсайту;
- тестування розробленого програмного забезпечення для збирання даних з вебсайту.

13

Рисунок В.13 – Слайд 13

Висновки

Проаналізовано загальні поняття про проблему домашнього насильства. Досліджено та проаналізовано програмні засоби для допомоги постраждалим від насильства.

Розроблено програмне забезпечення для збирання даних з вебсайту. Для створення програмного забезпечення збирання даних з вебсайтів обрано мову програмування Python, яка являє собою об'єктно-орієнтовану мову програмування високого рівня з вбудованими структурами даних і динамічною семантикою, підтримує кілька парадигм програмування, таких як структурне, об'єктно-орієнтоване та функціональне програмування, а також має великий набір бібліотек для аналізу вебданих. В якості середовища розробки для створення програмного забезпечення збирання даних з вебсайтів обрано Visual Studio Code, що є безкоштовним програмним забезпеченням, містить потужні функції та інструменти середовища розробки, підтримує широкий спектр мов програмування.

Результати тестування програмного забезпечення збирання даних з вебсайтів підтвердили його працездатність.

14

Рисунок В.14 – Слайд 14