Автоматизовані засоби парсингу проектів дистанційного навчання в Cloud для заповнення сайту http://dl-cloud.kpi.ua

Виконав: ст. гр. ДА-22
Ханенко О. А.
Керівник: доц., к. т. н.
Цурін О. П.

Мета роботи

1. Дослідження методів аналізу html документа

2. Аналіз існуючих методів завантаження веб-сторінок

3. Створення ПЗ парсингу веб-сайтів, для подальшого розміщення даних на http://dl-cloud.kpi.ua

Актуальність задачі

- Підтримка актуальності використовуваної на сайті інформації.
- Необхідність в об'єднанні інформації, так як вона перебуває на різних веб-сайтах.
- Заповнення сайту великими обсягами контенту.
- Забезпечення частого оновлення інформації на веб-сайті.

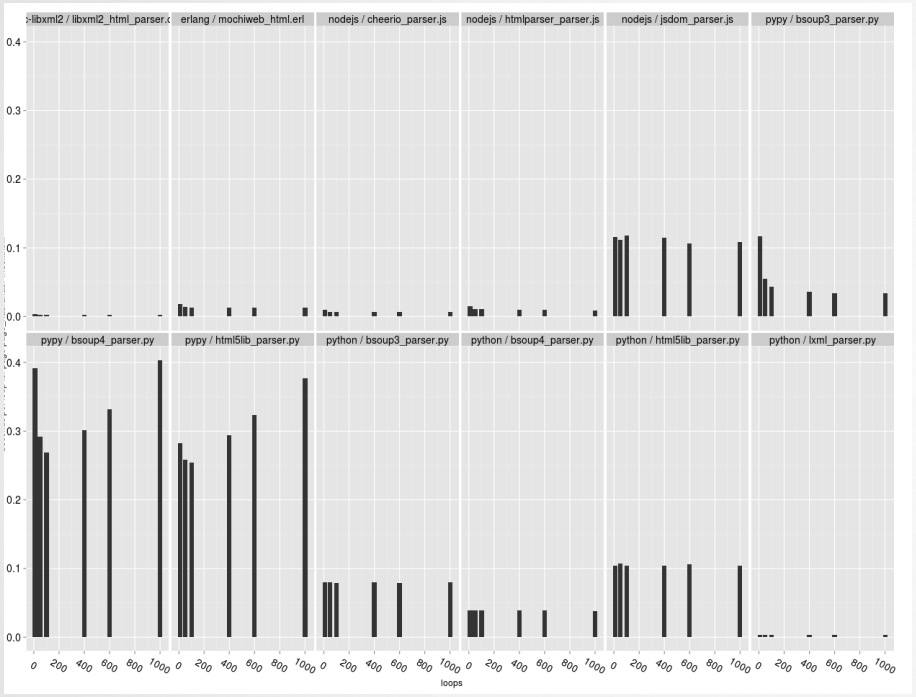
У порівнянні з людиною, програма-парсер:

- Швидко обійде тисячі веб-сторінок.
- Безпомилково відокремить потрібну інформацію від зайвої.
- Ефективно упакує кінцеві дані.

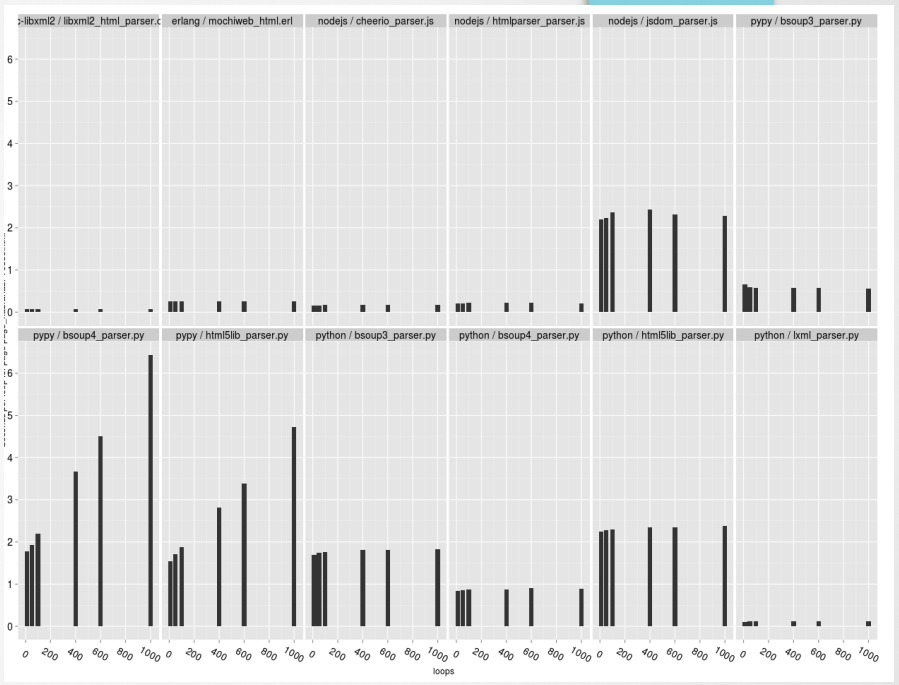
Бенчмарк для аналізу html

- Erlang — mochiweb_html
- Python — lxml.etree.HTML
- Python — BeautifulSoup 3
- Python — BeautifulSoup 4
- Python — html5lib
- PyPy — BeautifulSoup 3
- PyPy — BeautifulSoup 4
- PyPy — html5lib
- Node.JS — cheerio
- Node.JS — htmlparser
- Node.JS — jsdom
- C — libxml2

Залежність часу на обробку документа від числа ітерацій

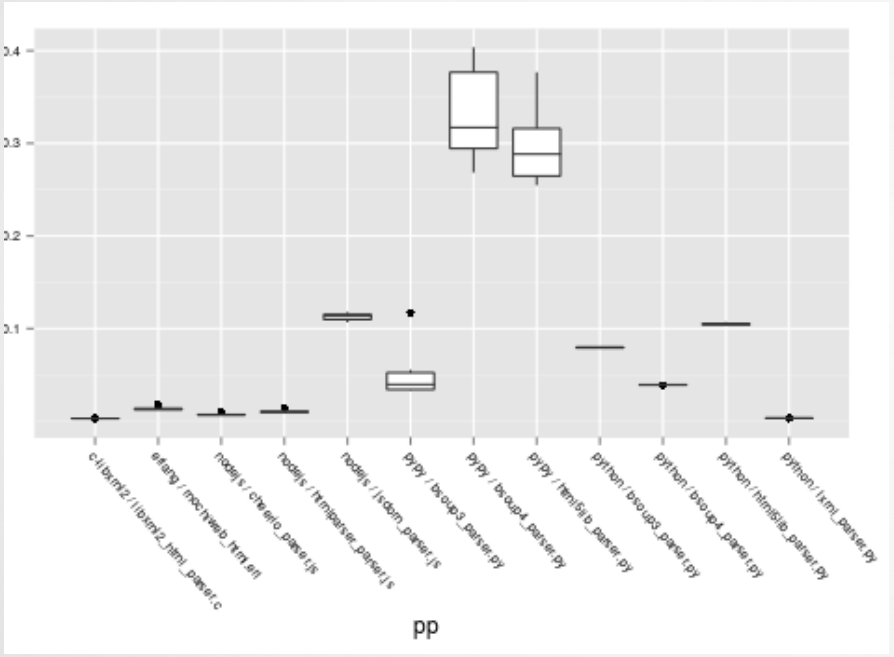


Файл розміром 95Кб

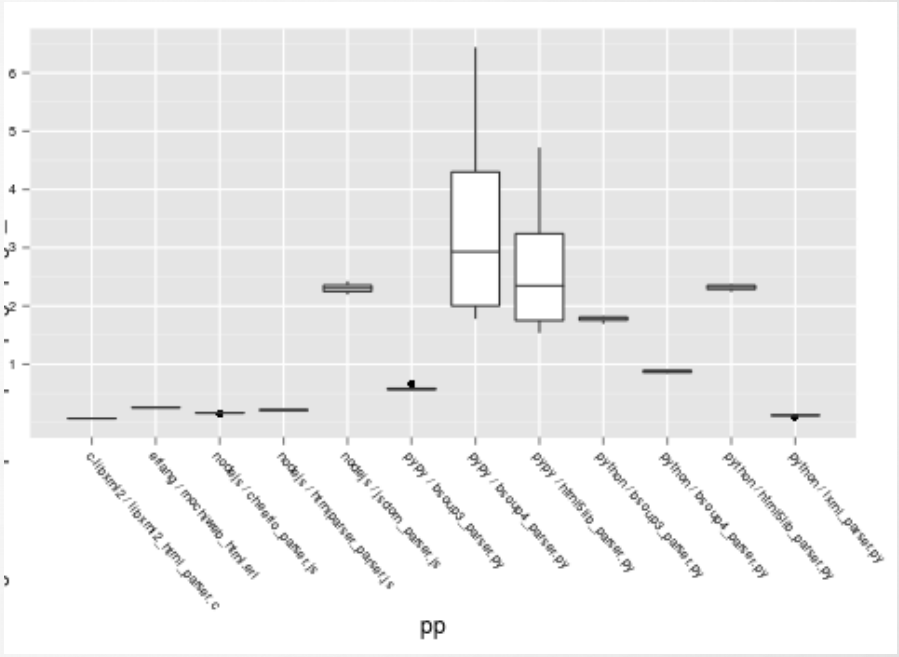


Файл розміром 1.6Мб

Середній час на обробку документа box-plot діаграма

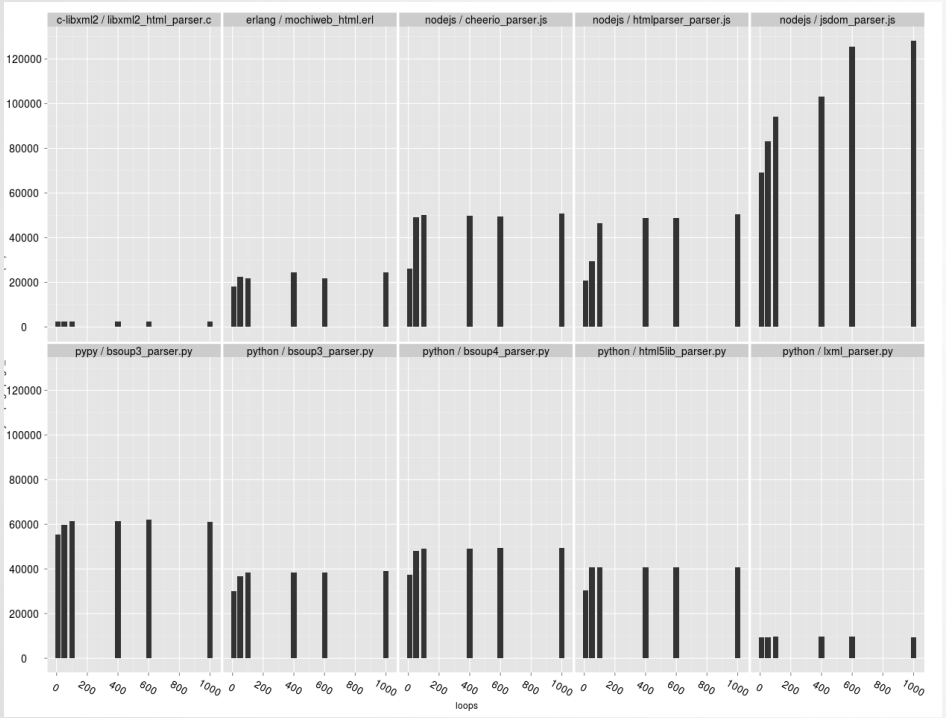


Файл розміром 95Кб

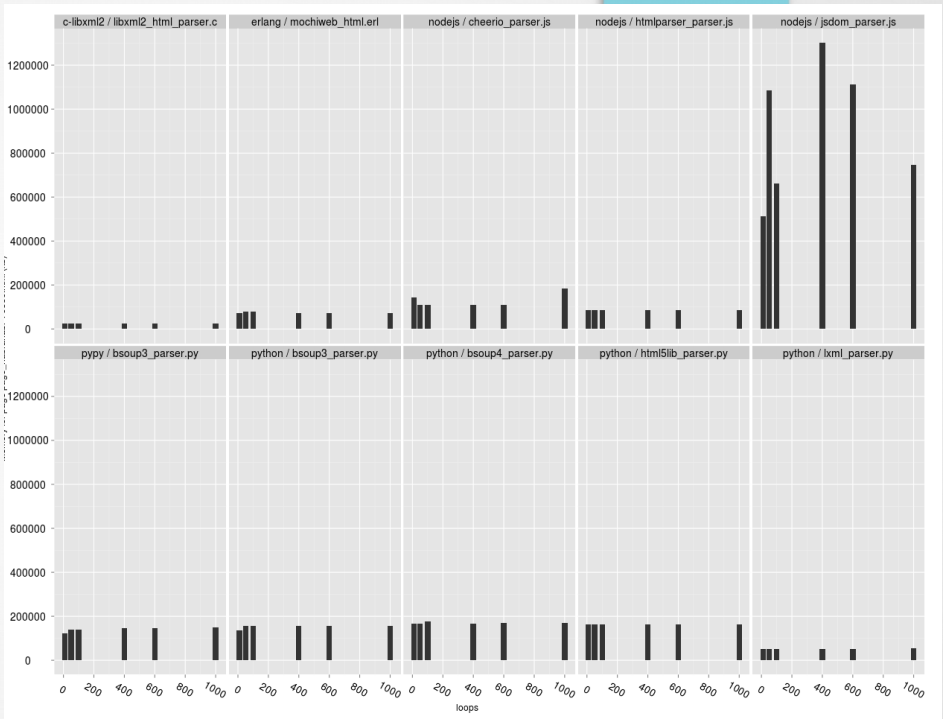


Файл розміром 1.6Мб

Споживання пам'яті в залежності від числа ітерацій парсеру.

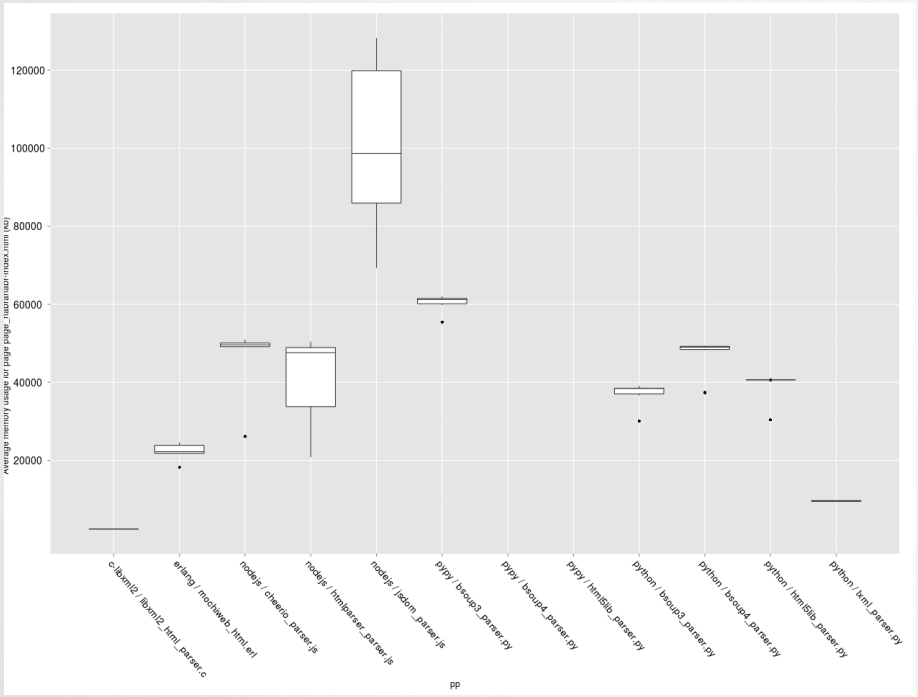


Файл розміром 95Кб

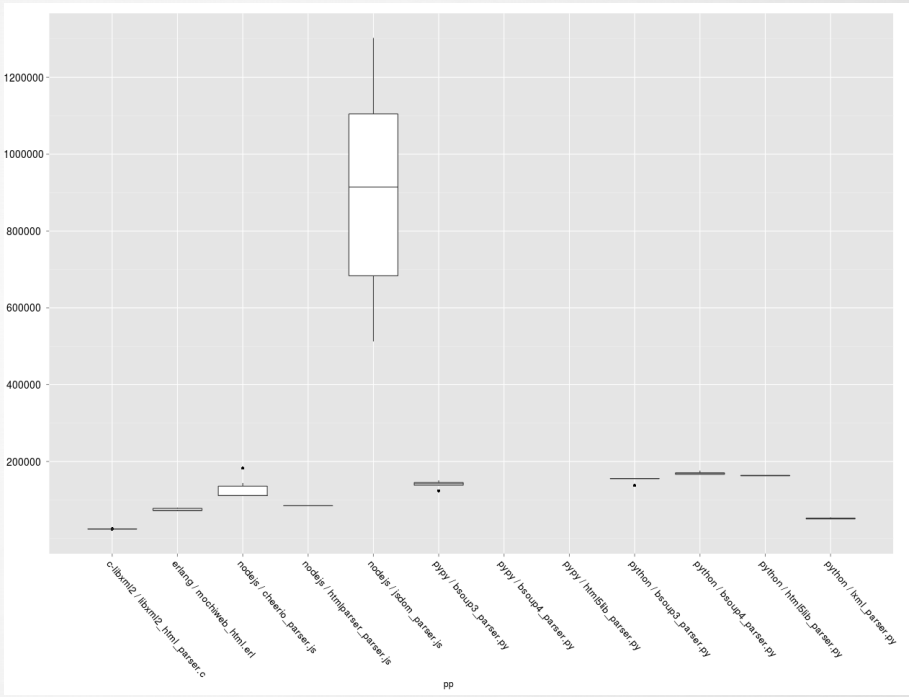


Файл розміром 1.6Мб

Усереднене споживання пам'яті box-plot



Файл розміром 95Кб



Файл розміром 1.6Мб

Переваги вибраного рішення

Python — lxml.etree.HTML

- Швидка і гнучка бібліотека для обробки XML/HTML на Python.
- Надає доступ до декларативного синтаксису XPath, зберігаючи при цьому загальну функціональність, доступну в Python
- Простота в написанні коду.
- Висока продуктивність при обробці дуже великих обсягів XML/HTML-даних.

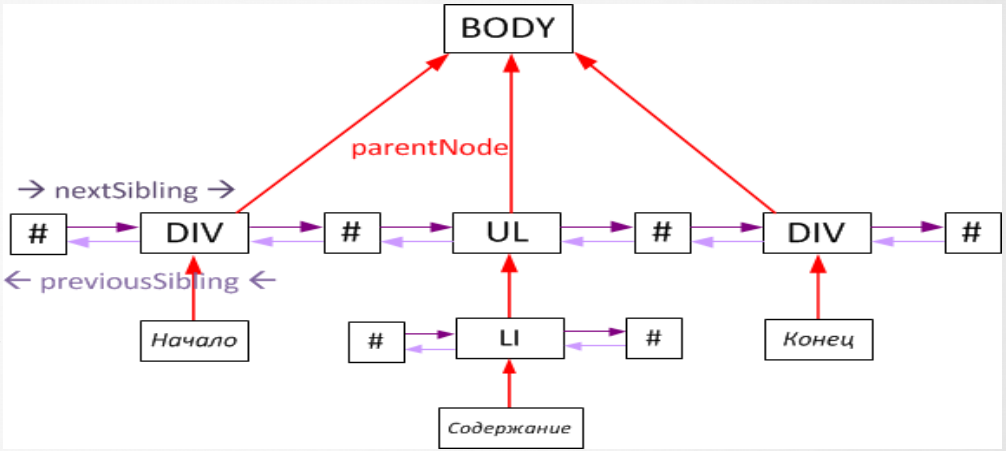
Дерево DOM

- **Об'єктна модель документа (DOM)— структура, яка представляє з себе деревовидне відображення структури XML/HTML документів.**

  У DOM виділено два типи вузлів.

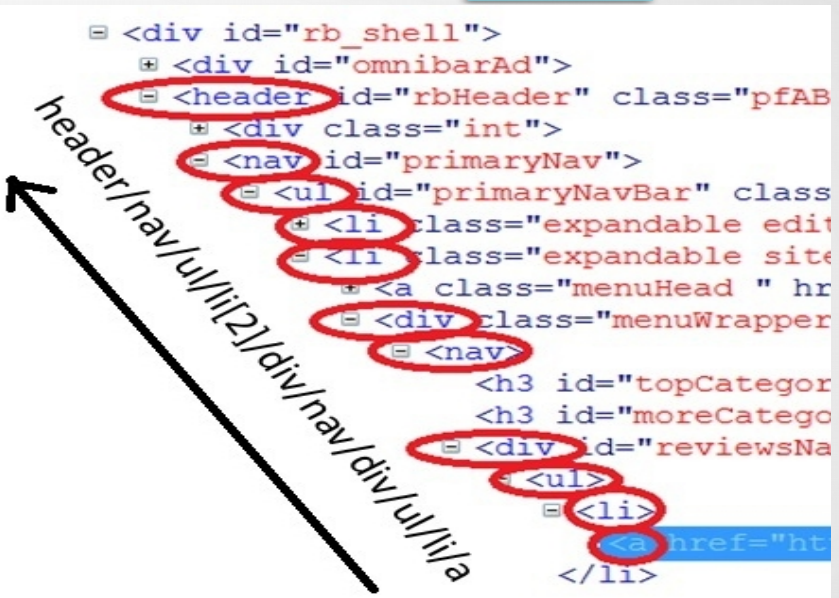
- ***Теги утворюють вузли-елементи (element node). Природним чином одні вузли вкладені в інші. Структура дерева утворена виключно за рахунок них.***

- ***Текст всередині елементів утворює текстові вузли (text node), позначені як #text. Текстовий вузол містить виключно рядок тексту і не може мати нащадків, тобто він завжди на самому нижньому рівні.***



Специфікація Document Object Model XPath

- Забезпечення функціональних можливостей доступу до DOM використовуючи Xpath
- lxml.etree включає в себе клас ElementTree, що надає метод xpath() для підтримки синтаксису Xpath, а також розширені функції.



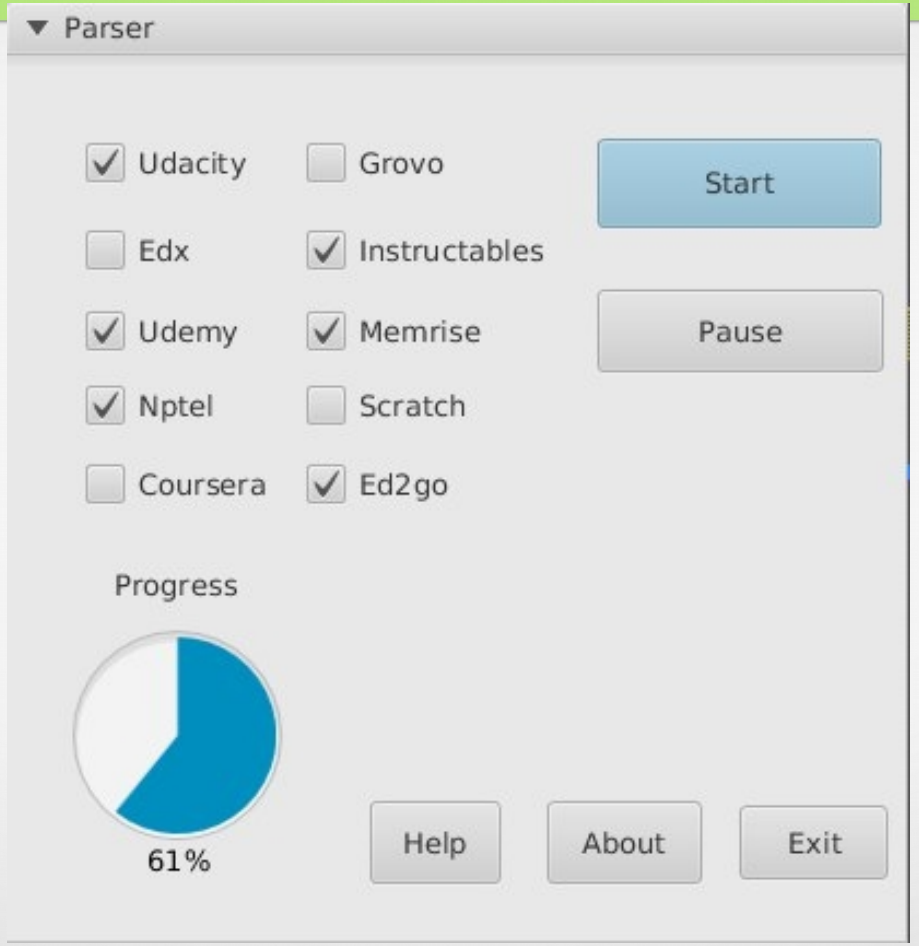
Завантаження веб-сторінок

urllib, urlib2

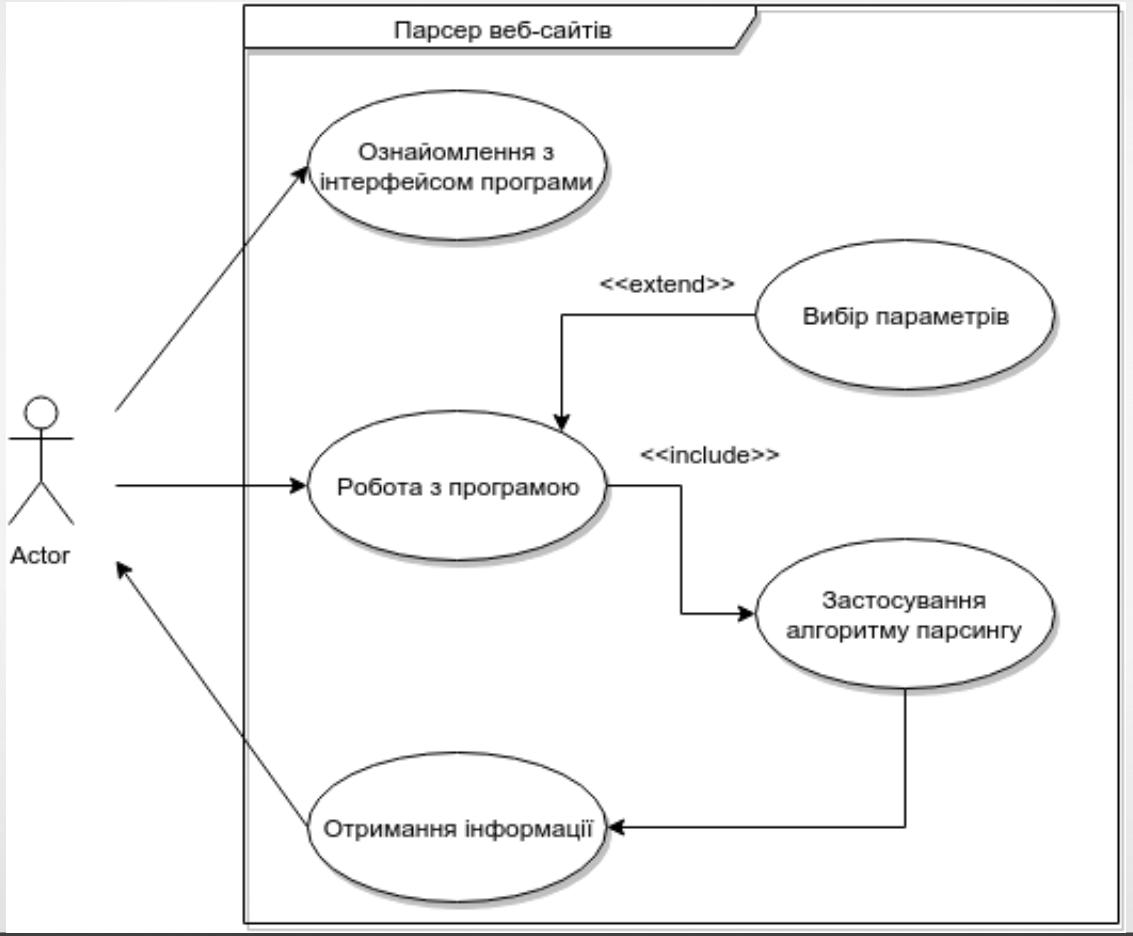
- Повторна спроба завантаження
- Налаштування агента користувача
- Підтримка проксі-сервера
- Веб-сторінки з динамічним контентом
- Взаємодія з формами
- Обхід CAPTCHA

- Download() повторно викликається у разі помилки 5.xx
- User-agent: Mozilla/5.0
- urllib2.ProxyHandler(proxy_params)
- Selenium (WebDriver) send.keys()
- Mechanize
- Pillow + Tesseract,

GUI


Parser
Udacity
Grovo
Start
Edx
Instructables
Udemy
Memrise
Pause
Nptel
Scratch
Coursera
Ed2go
Progress
61%
Help
About
Exit

Діаграма прецедентів



Результати роботи

	A	B	C
1	lessons_name	lessons_url	category
2	How to Use Git and GitHub	https://www.udacity.com/course/how-to-use-git-and-github--ud775	software-engineering
3	Intro to Computer Science	https://www.udacity.com/course/intro-to-computer-science--cs101	software-engineering
4	Intro to Java Programming	https://www.udacity.com/course/intro-to-java-programming--cs046	software-engineering
5	Programming Foundations with Python	https://www.udacity.com/course/programming-foundations-with-python--ud036	software-engineering
6	Intro to Hadoop and MapReduce	https://www.udacity.com/course/intro-to-hadoop-and-mapreduce--ud617	software-engineering
7	Software Development Process	https://www.udacity.com/course/software-development-process--ud805	software-engineering
8	Artificial Intelligence for Robotics	https://www.udacity.com/course/artificial-intelligence-for-robotics--cs373	software-engineering
9	Data Wrangling with MongoDB	https://www.udacity.com/course/data-wrangling-with-mongodb--ud032	software-engineering
10	Intro to Parallel Programming	https://www.udacity.com/course/intro-to-parallel-programming--cs344	software-engineering
11	Intro to Artificial Intelligence	https://www.udacity.com/course/intro-to-artificial-intelligence--cs271	software-engineering
12	Interactive 3D Graphics	https://www.udacity.com/course/interactive-3d-graphics--cs291	software-engineering
13	Software Testing	https://www.udacity.com/course/software-testing--cs258	software-engineering
14	Design of Computer Programs	https://www.udacity.com/course/design-of-computer-programs--cs212	software-engineering
15	Intro to Algorithms	https://www.udacity.com/course/intro-to-algorithms--cs215	software-engineering
16	Programming Languages	https://www.udacity.com/course/programming-languages--cs262	software-engineering
17	Applied Cryptography	https://www.udacity.com/course/applied-cryptography--cs387	software-engineering
18	Software Debugging	https://www.udacity.com/course/software-debugging--cs259	software-engineering
19	Intro to Theoretical Computer Science	https://www.udacity.com/course/intro-to-theoretical-computer-science--cs313	software-engineering
20	Machine Learning: Unsupervised Learning	https://www.udacity.com/course/machine-learning-unsupervised-learning--ud741	software-engineering
21	Networking for Web Developers	https://www.udacity.com/course/networking-for-web-developers--ud256	software-engineering
22	Scalable Microservices with Kubernetes	https://www.udacity.com/course/scalable-microservices-with-kubernetes--ud615	software-engineering
23	Technical Interview	https://www.udacity.com/course/technical-interview--ud513	software-engineering
24	How to Build a Startup	https://www.udacity.com/course/how-to-build-a-startup--ep245	non-tech
25	Product Design	https://www.udacity.com/course/product-design--ud509	non-tech
26	App Monetization	https://www.udacity.com/course/app-monetization--ud518	non-tech
27	Rapid Prototyping	https://www.udacity.com/course/rapid-prototyping--ud723	non-tech
28	App Marketing	https://www.udacity.com/course/app-marketing--ud719	non-tech
29	Statistics	https://www.udacity.com/course/statistics--st095	non-tech
30	Intro to the Design of Everyday Things	https://www.udacity.com/course/intro-to-the-design-of-everyday-things--design101	non-tech

data

Висновки

- Проаналізовані підходи парсингу, вибрано оптимальний варіант.
- Створено ПЗ парсингу веб-сайтів.
- Отримано результати роботи програми.

Дякую за увагу