



**UNIVERSITATEA DE VEST DIN TIMIȘOARA**  
**FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ**  
**INFORMATICA APLICATA**  
**Subgrupa-5**

# **Documentație**

## **News Spider**

**~Java Web Scraper ~**

***Proiect Individual 2021-2022***

**Student: Madaras Andrei – Iulian**

Timișoara

11/2021

## Cuprins

Introducere .....	3
Capitolul 1. ....	4
Capitolul 2. ....	5
2.1. Funcționalitățile aplicației.....	5
2.2. Dependente si tehnologii folosite.....	6
2.3 Configurația sistemului .....	7
Capitolul 3. Functionalitate:.....	9
Manual de Utilizare.....	21
Capitolul 4. ....	37
Capitolul 5. ....	38
Capitolul 6: .....	39
Grafic de Realizare al Proiectului : .....	40
Referinte Bibliografice. ....	42

## Introducere

Webcrawler-ele/Webscraper-ele au ramas o constanta in "landscape-ul" internetului, inasa odata cu avansul tehnologic si metodele de detectare ale oricarui bot facand si simpla preluare de date foarte dificila, fiind nevoie de proxy-uri(adrese Ip intermediare, pentru a nu fi blacklist-uit pe Ip-ul adevarat) in numar mare, simulatoare de browser pentru a putea evoca javascriptul unor pagini,cat si multe altele.

Astfel vreau sa ma provoc in vederea realizarii unui astfel de proiect de mare complexitate, pornind de la o forma simpla/demonstrativa (un "proof of concept") al webcrawlerul-ui si adaugand functionalitati pe parcursul dezvoltarii proiectului .

Doresc sa implementez un Web Crawler/Scraper cu o interfata interesanta unde fiecare nod reprezinta o adresa catre o pagina ceruta dintr-o lista create de catre utilizator pt a obtine informatii repede despre un anumit subiect ( putand cauta cuvinte cheie (search words) si gasii fraze, propozitii sau imaginii ce au legatura cu cerinta acestuia) , acesta putand sa "interogheze" serverele paginilor respective pt informatia ceruta fara a fi detectat.

Potentiali Utilizatori: News Spider difera fata e alte solutii de acest tip prin interfata simpla si usor de inteles pe care o ofera, astfel oricine poate utiliza aplicatia in scopul adunarii de date ce-l pot interesa de pe anumite pagini web.

## Capitolul 1.

Web scrapere/Web cawlere sunt folosite de catre orice motor de cautare, linkuri/imagini sau text prezentate de catre un browser relative la o pagina sau domeniu sunt toate obtinute prin intermediul utilizarii unor roboti de indexare cat si a unora de parsare a continutului paginii, In plus foarte multe companii de marketing sau ad service utilizeaza solutii deja existente oferite de catre retaileri precum “WebscraperApi” sau custom made pe framework-uri precum “scrapy”.

Mai exista si cawlere specializate precum cele de “sentiment analysis” utilizate de investitori pentru prezicerea starii pietelor monetare, de actiuni sau crypto, cat si alte solutii ce utilizeaza concepte de Machine learning.

O Lista de crawler-i recunoscuti de site-ul de comert pcgarage.ro

```
User-agent: Exabot
User-agent: AhrefsBot
User-agent: 007ac9
User-agent: SISTRIX Crawler
User-agent: Scrubby
User-agent: Robozilla
User-agent: MJ12bot
User-agent: SEOkicks-Robot
User-agent: UptimeRobot
User-agent: Ezooms Robot
User-agent: WiseGuys Robot
User-agent: Turnitin Robot
User-agent: Heritrix
User-agent: magpie-crawler
User-agent: CCBot
User-agent: SentiBot
User-agent: Cliqzbot
User-agent: Mediatoolkitbot
User-agent: MegaIndex
User-agent: FatBot
User-agent: Curious George
User-agent: TinEye-bot
User-agent: VegeBot
User-agent: Baiduspider
User-agent: Baiduspider-image
User-agent: YisouSpider
User-agent: proximic
User-agent: SpeedySpider
```

News Spider se deosebeste de alte solutii prezente pe piata prin interfata usor de utilizat si interesant de vizualizat

## Capitolul 2.

### 2.1. Funcționalitățile aplicației

- Alcatuirea unei liste de url-uri care vor fi scrape-uite de informatii dupa un KeyWord dat:
- Separare in functie de cat de greu este sa extragi datele (“ safe “ page vs “ non -safe” page
- Meniu de optiuni in care utilizatorul va putea alege:
  - Nr de threaduri care pot fi utilizate pt a accelera procesul
  - Optiuni in functie de pastrarea fisierelor temporare ( lista de url-uri, log-uri, .json) ce continue rezultate-le parsingului paginii
  - Rata “de default” la care se va face requestul ( aceasta va fi comparata cu una reala a paginii prin parsarea fisierului robots.txt care exista pe aproape orice pagina)
  - Salvarea unui screencap al paginii
- Graf de noduri , unde nodurile vor fi url-urile sau/si domeniile scrape-uite / crawluite sub forma cat mai apropiata de a unei panze de paianjen.
  - Prin interactiunea cu nodurile utilizatorul va putea vedea datele obtinute de la paginile respective.

## 2.2. Dependente si tehnologii folosite

### **Limbaj de programare:**

NewsSpider este scris in Java.

### **De ce Java?:**

Java ofera scalabilitate mai mare decat Python sau C/C++/C#, fara a faulta intelegerea pe care o am asupra modului de creare a unei astfel de aplicatii.

### **Librarii:**

-Jsoup

- ConcurrentFutures si Threading ( pentru utilizarea multithreadingului in aplicatie).

-Requests(pentru trimiterea de requesturi ).

- Swing si AWT (librarii standard de design de GUI in Python).

-Librarii standard din Java ( precum: random , time , math , re)

-Jackson

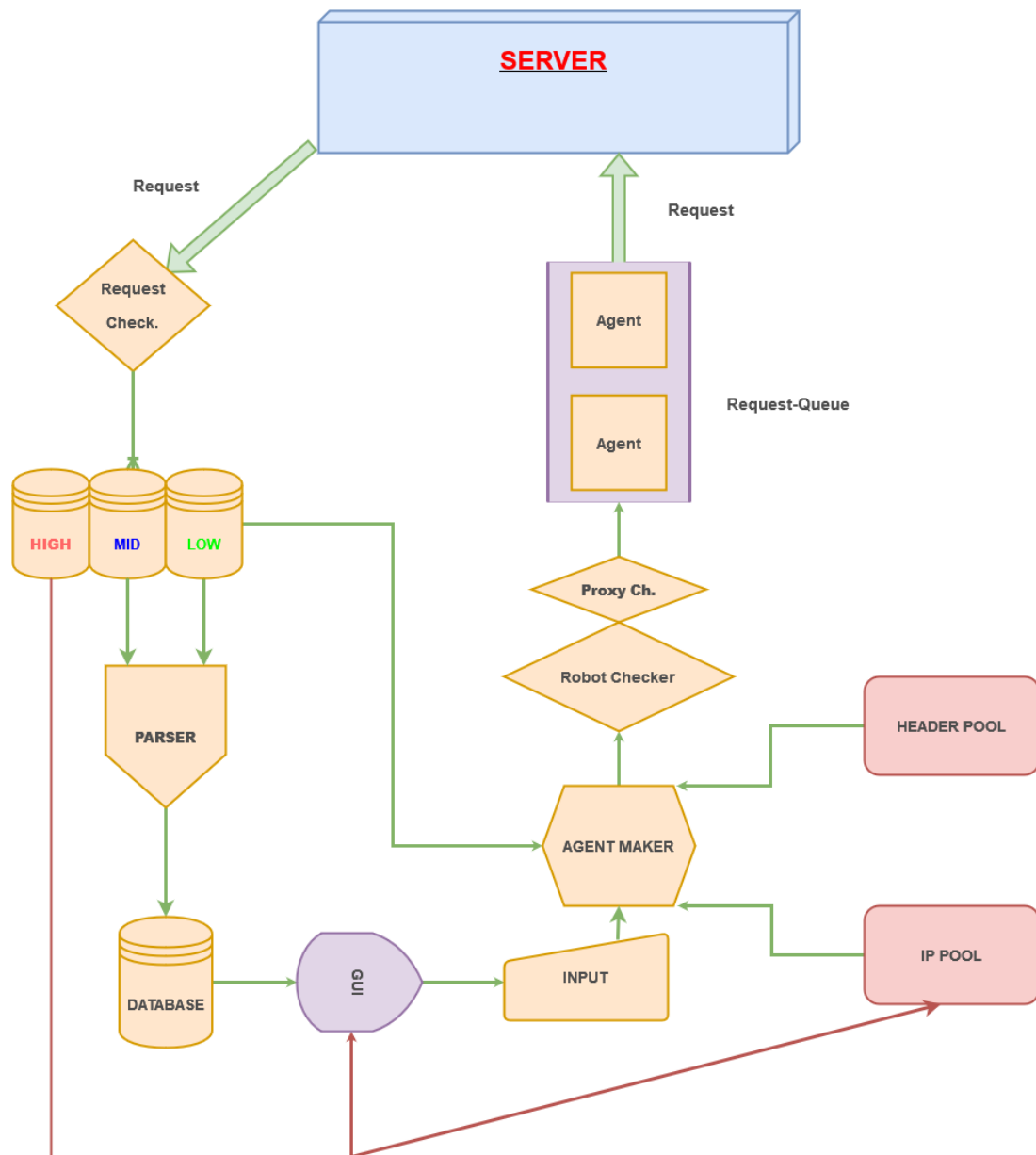
-Pair

### **API/Framework:**

-Playwright(Simulator de instanta de browser)

## 2.3 Configurația sistemului

### Schema UML Initiala:



Acesata este schema UML de la care am pornit urmand sa modific

- **Cum Functioneaza?:**

Webcrawler-ul va utiliza o varietate de librarii pentru a extrage date din anumite site-uri web(date dinainte, sau cautate in functie de un key-word/tag), ca apoi sa stocheze un "rezumat" intr-o baza de date locala/externa care va putea fi vizualizat in GUI.

Voi utiliza o clasa de tip "Agent" care va contine:

- Header-ul requestului(informatii despre browser, sistem de operare , geolocalie , data la care a fost facut requestul , etc.)
- Instanta de playwright care va rula requestul daca este necesar( WebGL check. , javascript care adauga elemente paginii)
- Frecventa la care se vor face requesturile asociate unui "Agent" ( pentru a da aparenta unui utilizator uman voi randomiza dupa intervale , voi alege frecventa mica pentru a nu atinge rate-limit-ul serverului)

Agentul instantiat va fi asignat unui thread pentru a trimite requestul

In functie de raspunsul serverului avem 3 posibilitati:

- Eroare ( 301, 404, etc.) ( Vom trimite request-ul din nou un anumit numar de ori pana cand vom scoate instanta de Agent din queue)
- Acces(Vom primi continutul paginii si o v-om trimite mai departe la parser/scraper , iar adresa o v-om pune intr-o lista de url-uri)

Informatii-le parsate v-or fi salvate local/extern si aduse la GUI.

Nu mai utilizez " IP POOL " deoarece am access doar la putine ip-uri rezidentiale care le pot folosii ca si proxy-uri, astfel nu voi mai trece prin "Proxy Check".



## Capitolul 3. Functionalitate:

Aplicatia detine in alcaturie mai multe meniuri prin care utilizatorul poate sa interactioneze cu aplicatia cat si cu procesul de cautare prin inserarea de optiuni in meniul de optiuni, prin inserarea sau stergerea de URL-uri in meniul denumit "URLs", prin interactiunea cu baza de date locala in care sunt salvate mai multe vederi ale unor date salvate , utilizate si extrase in urma si la baza procesului de cautare .

Toate meniurile , cu exceptia celor pentru interactiunea cu baza de date sunt resizable, si nu pot exista mai multe instante ale GUI-ului(precum ar fi problema pentru GUI-ul asociat interactiunii cu baza de date)

### Meniul Principal



## Lista de adrese URL

News Spider Test

Meniu

URL LIST

URLs in list

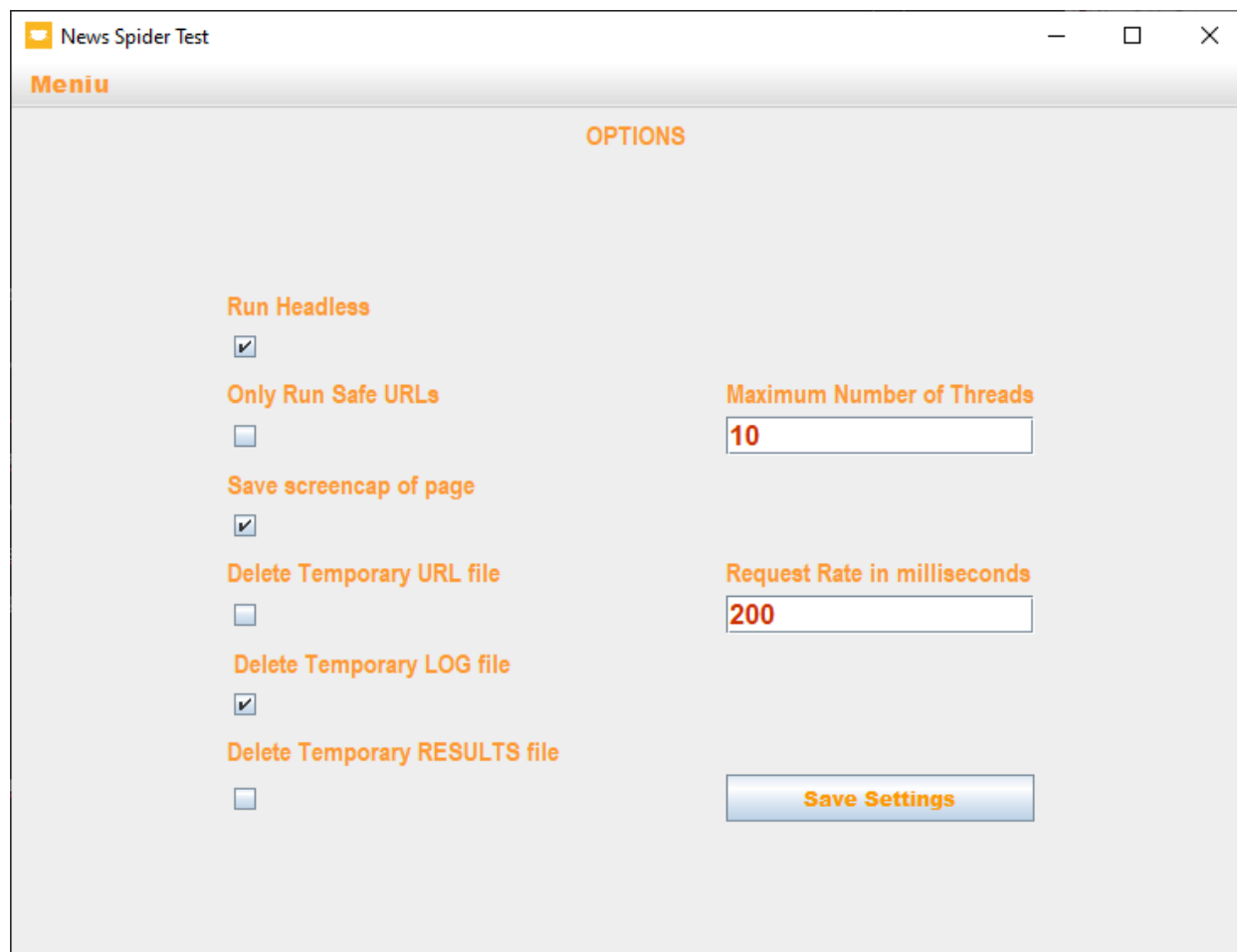
- https://libertatea.ro
- https://www.theguardian.com
- https://www.theverge.com
- https://www.reuters.com
- https://emag.ro
- https://www.exemplu.com

Insert or Delete Valid URL

ADD

DELETE

Se pot insera/sterge/ copia linkuri (de format : https:// altfel va apareea un pop-up ce va instiinta utiliztorul despre incalcarea formatului), chiar si in timpul utilizarii a altor facilitate sau a rularii proceselor “din backend”, urls sunt salvate intr-un fisier de tip json avand un atribut privind dificultatea obtinerii datelor( safe url – passes all checks , unsafe – nu am putut accesa sau nu am putut sat rec de gdpr pop-up sau alte pop-upuri)



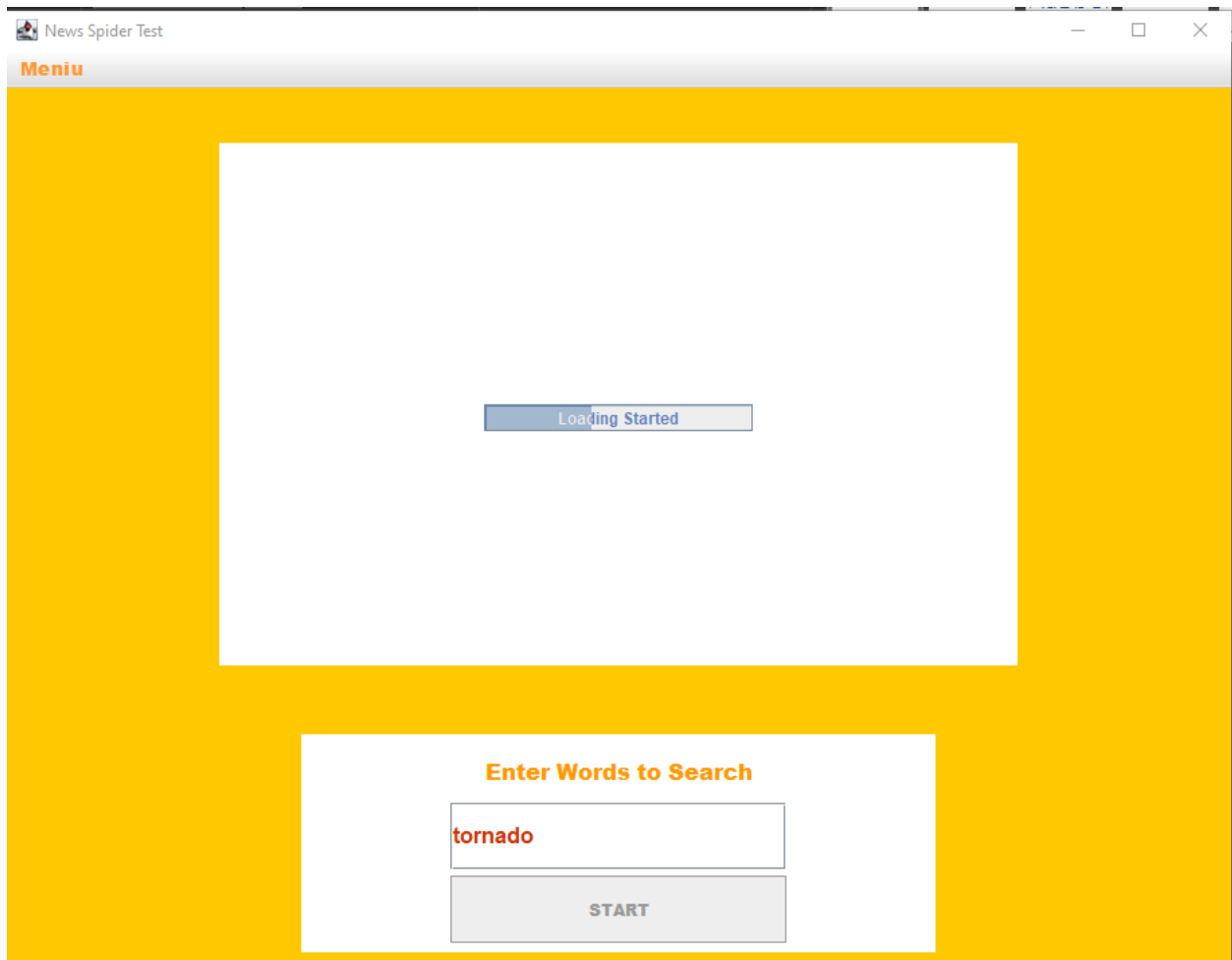
-Meniu de optiuni in care utilizatorul va putea allege:

- Nr de threaduri care pot fi utilizate pentru a accelera procesul
- Optiuni in functie de pastrarea fisierelor temporare( lista de url-uri, log-uri, json ce continue rezultate-le parsingului paginii
- Rata “de default” la care se va face requestul ( aceasta va fi comparata cu una reala a paginii prin parsarea fisierului robots.txt care exista pe aproape orice pagina)
- Salvarea unui screencap al paginii

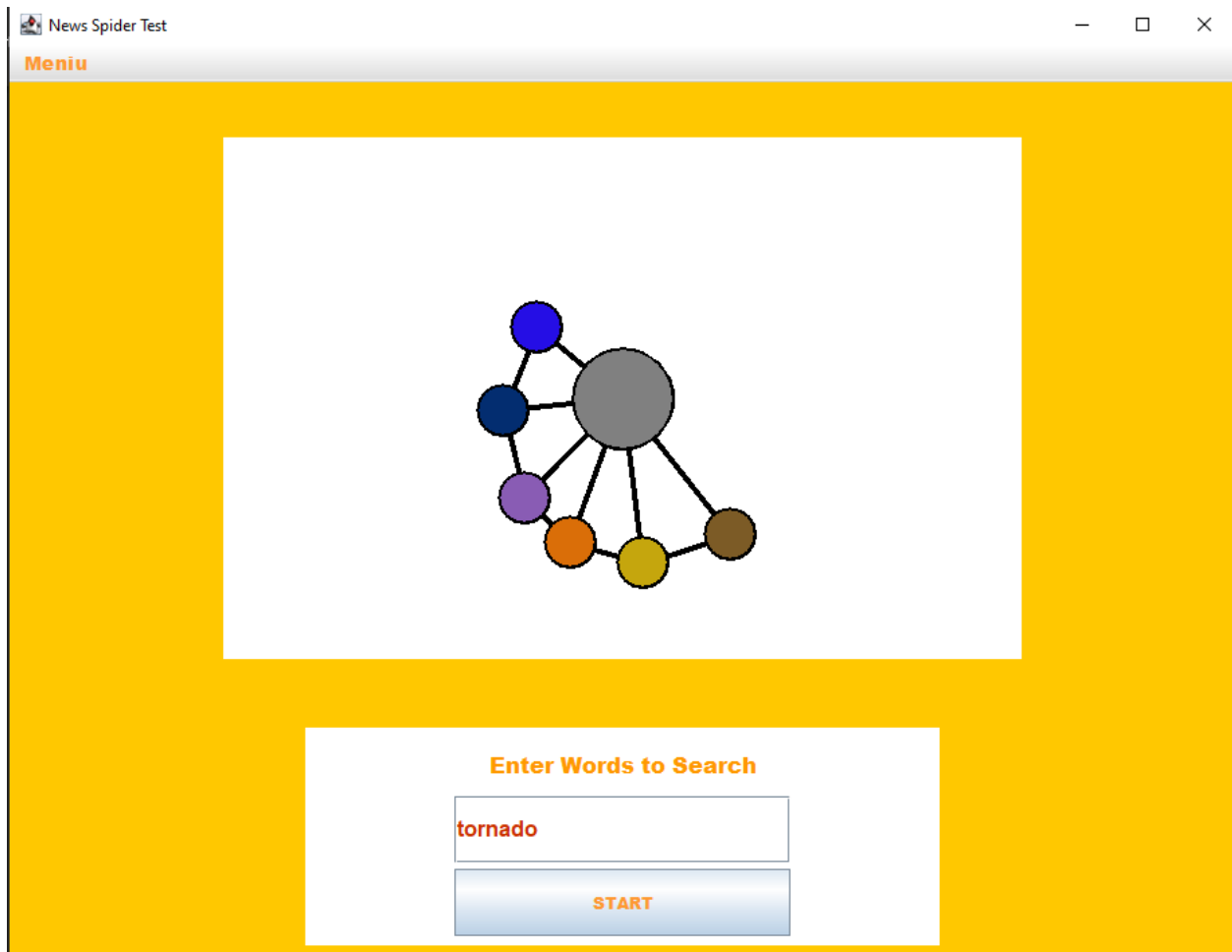
Optiunile sunt salvate intr-un fisier de tip .json in timpul rularii , se pot modifica in timpul rularii alor procese

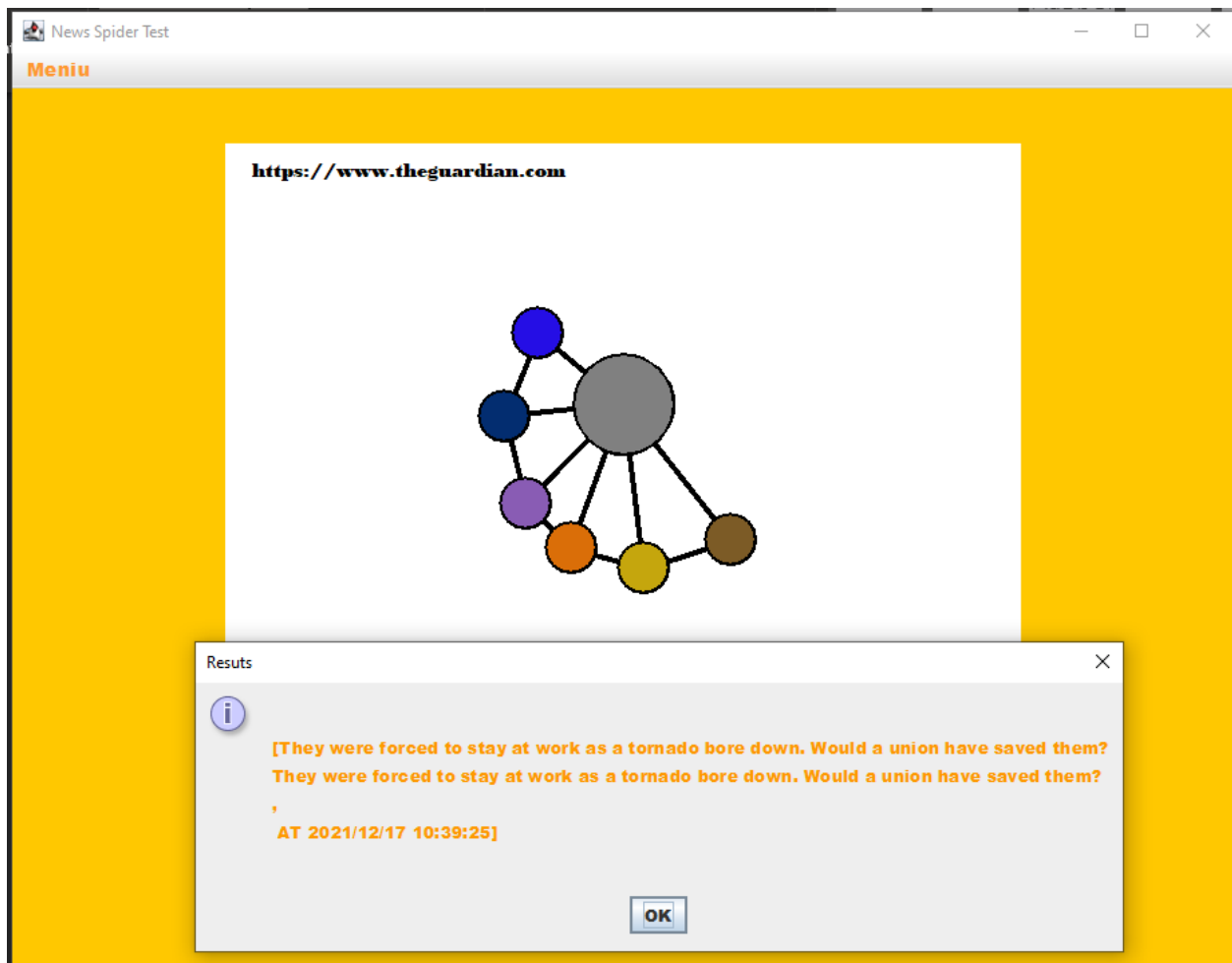
Graf de noduri , unde nodurile vor fi url-urile sau/si domeniile scrape-uite / crawluite sub forma cat mai apropiata de a unei panze de paianjen .

Prin interactiunea cu nodurile utilizatorul va putea vedea datele obtinute de la paginile respective prin intermediul unui "frame" adaugat gui-ului cat si data la care a fost obtinute si/sau daca au fost obtinute

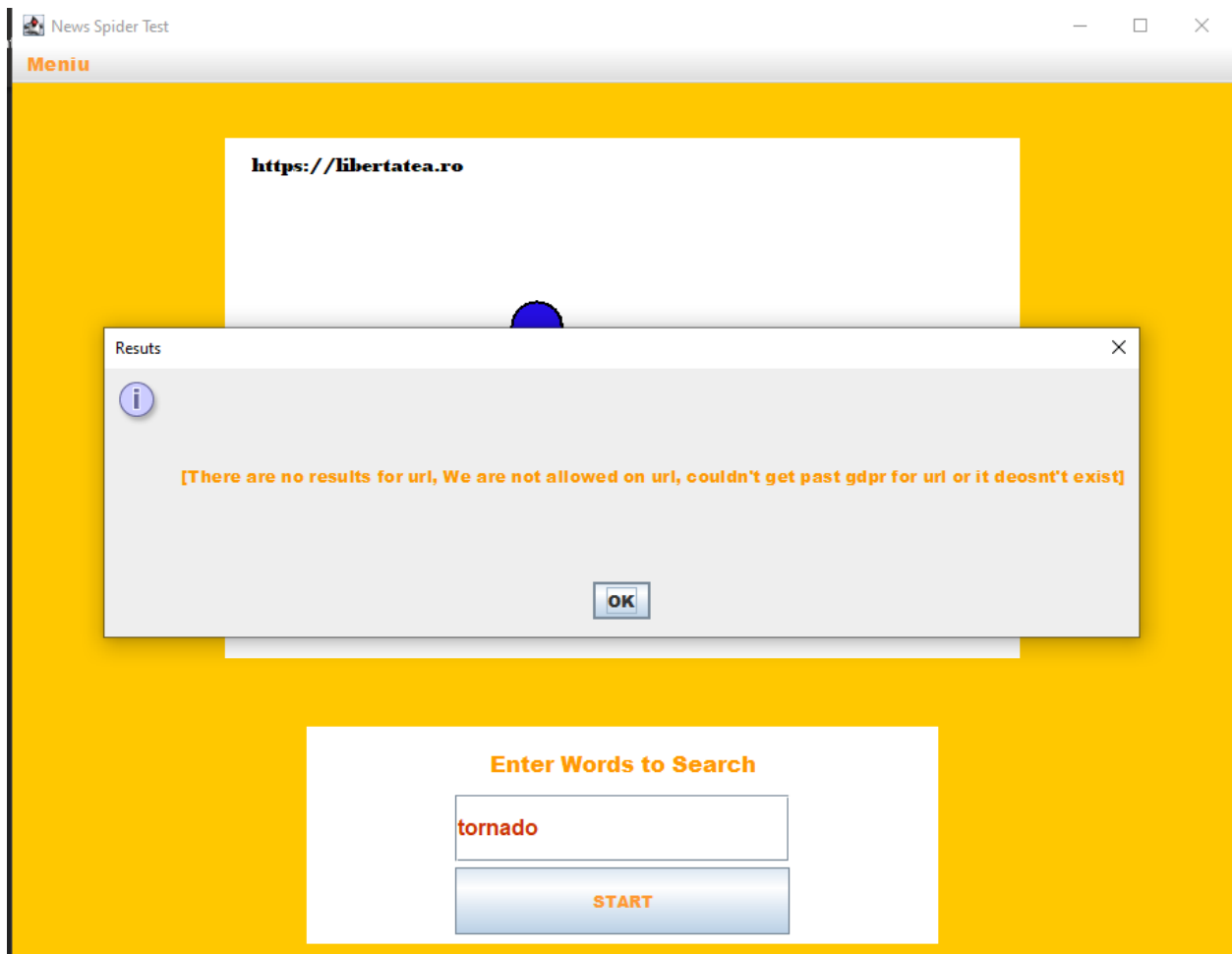


## Graful cu nodurile rezultate



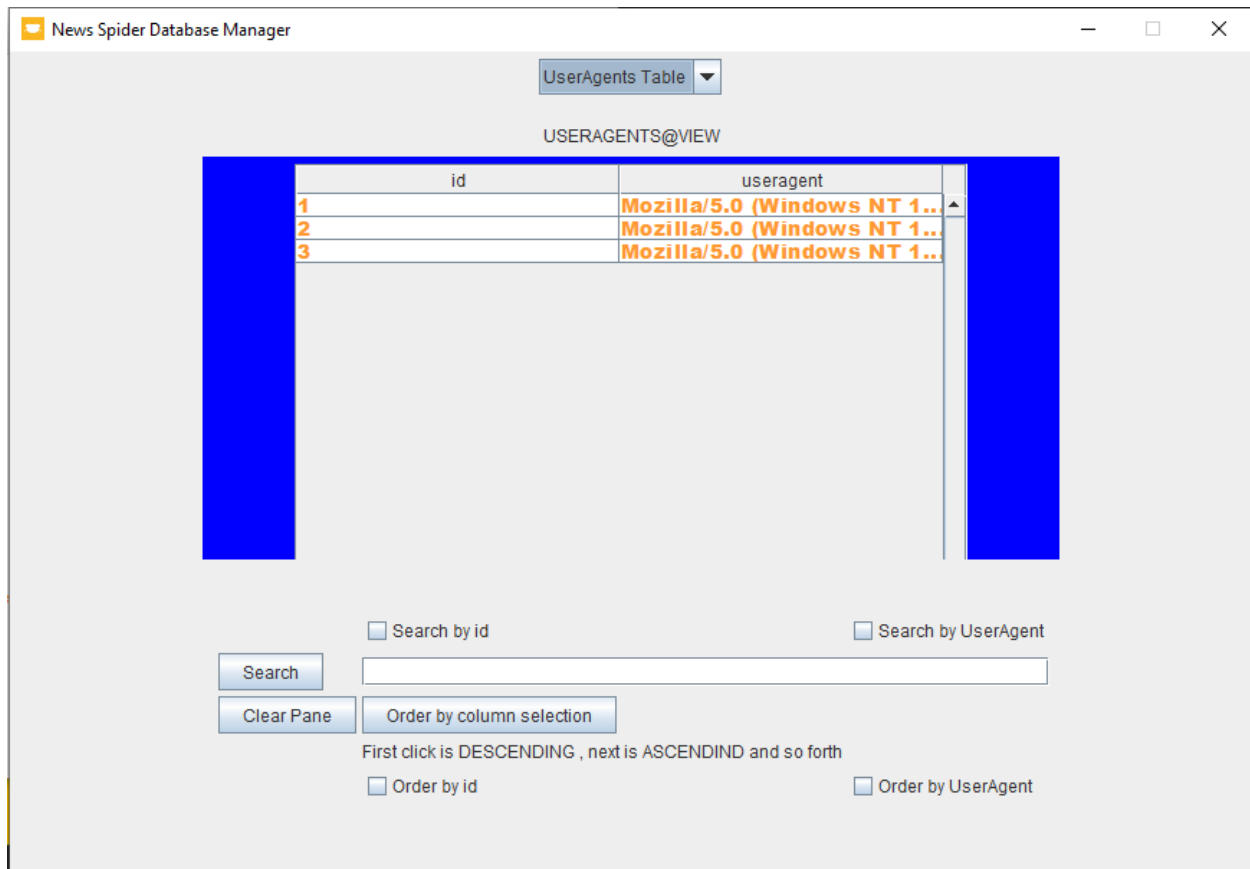


Se poate observa tot textul prezent pe pagina ce are legatura cu KeyWord-ul "tornado".



Pagina care nu contine text sau referinte la KeyWord-ul "tornado" , ori nu am putut sa extrag datele

## Vedere a Tabelului cu “User-Agents” din Baza de date



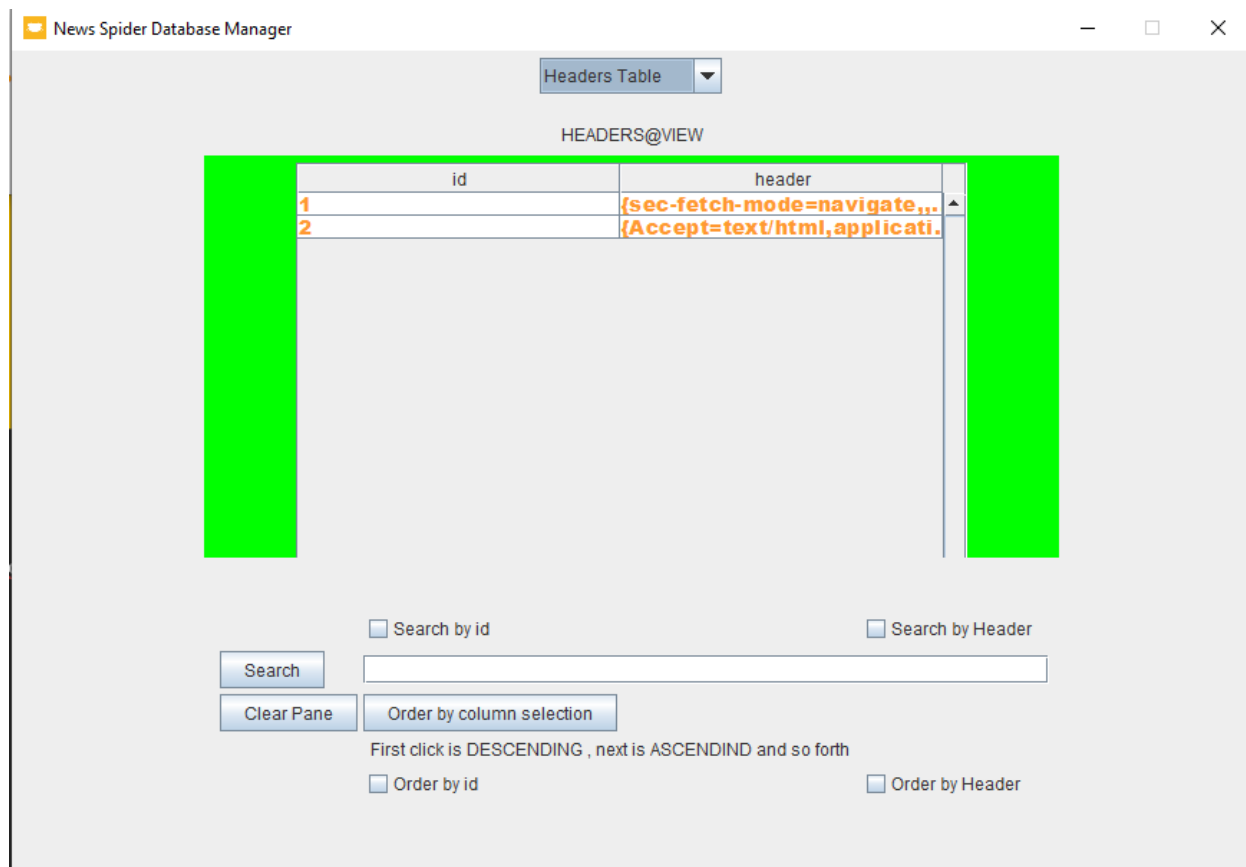
The screenshot shows the 'News Spider Database Manager' window. At the top, a dropdown menu is set to 'UserAgents Table'. Below it, the text 'USERAGENTS@VIEW' is displayed. A table with two columns, 'id' and 'useragent', is shown. The table contains three rows of data, all with the same useragent string. Below the table, there are search and sorting options. The search section includes a 'Search' button, a 'Clear Pane' button, and a search input field. There are checkboxes for 'Search by id' and 'Search by UserAgent'. The sorting section includes an 'Order by column selection' button and checkboxes for 'Order by id' and 'Order by UserAgent'. A note states: 'First click is DESCENDING , next is ASCENDIND and so forth'.

id	useragent
1	Mozilla/5.0 (Windows NT 1...
2	Mozilla/5.0 (Windows NT 1...
3	Mozilla/5.0 (Windows NT 1...

Tabelul detine pe cele doua coloane informatii privind identificare cat si stringul denominator al unui “user-agent” ce va fi utilizat in procesul de cautare.

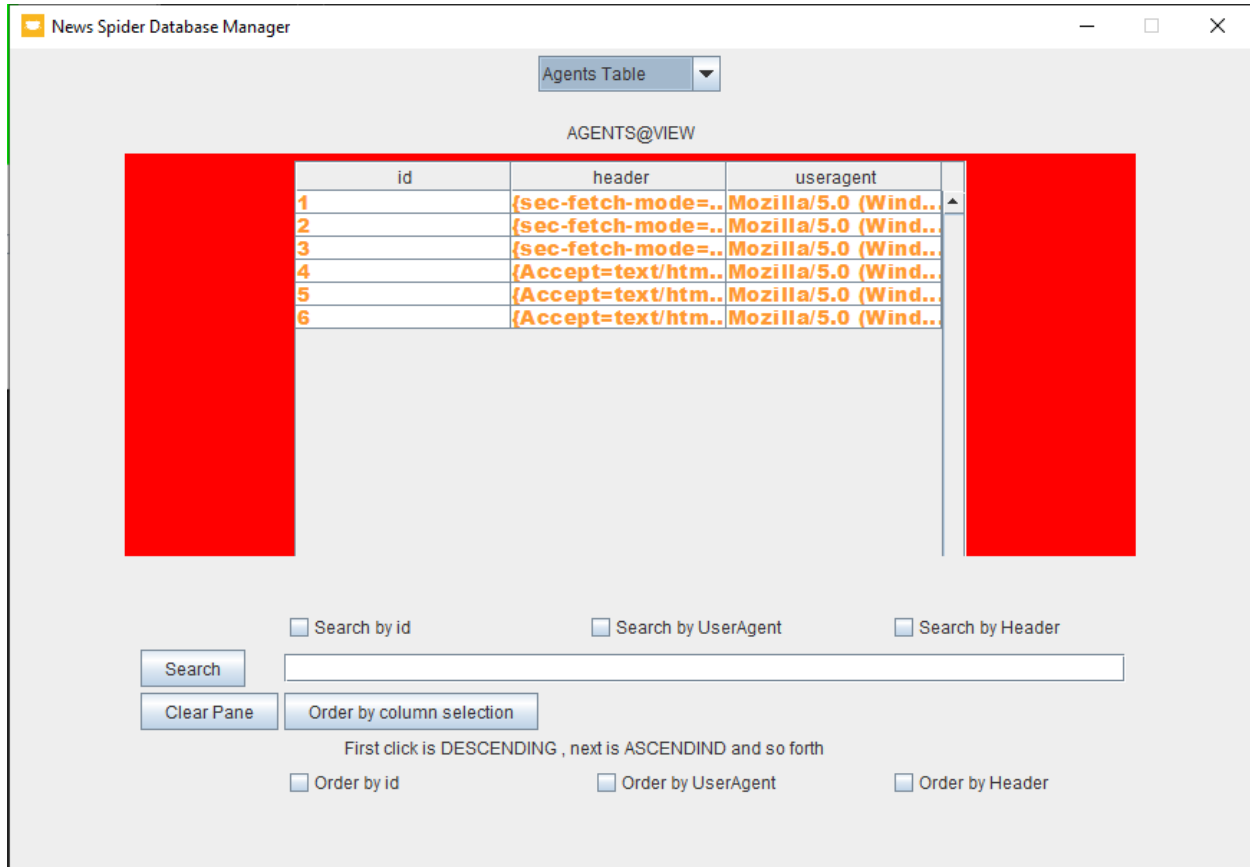


## Vedere a tabelului cu “Headere”



Tabelul detine pe cele doua coloane informatii privind identificare cat si stringul denominator al unui “Header” ce va fi utilizat in procesul de cautare.

## Vedere a tabelului cu “Agenti”



News Spider Database Manager

Agents Table

AGENTS@VIEW

id	header	useragent
1	{sec-fetch-mode=..	Mozilla/5.0 (Wind...
2	{sec-fetch-mode=..	Mozilla/5.0 (Wind...
3	{sec-fetch-mode=..	Mozilla/5.0 (Wind...
4	{Accept=text/htm..	Mozilla/5.0 (Wind...
5	{Accept=text/htm..	Mozilla/5.0 (Wind...
6	{Accept=text/htm..	Mozilla/5.0 (Wind...

☐ Search by id    ☐ Search by UserAgent    ☐ Search by Header

Search

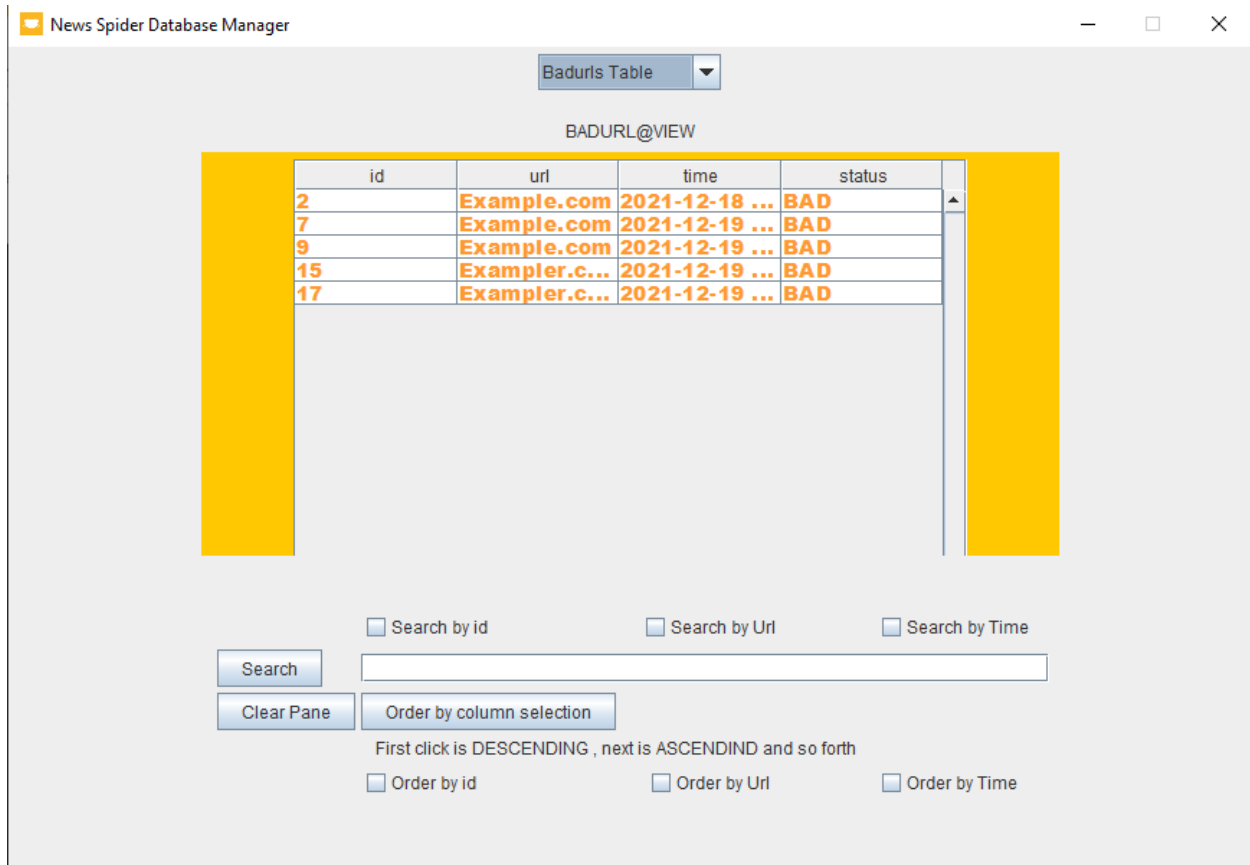
Clear Pane    Order by column selection

First click is DESCENDING , next is ASCENDING and so forth

☐ Order by id    ☐ Order by UserAgent    ☐ Order by Header

Tabelul detine pe cele trei coloane informatii privind identificare cat si combinatiile posibile de “User-Agent” si “Header” ce vor fi utilizate in “inserarea” intr-un agent (context de headless browser generat de Playwright)

## Vedere a tabelului ce continue adresele “ilegale”



The screenshot shows the 'News Spider Database Manager' interface. At the top, there's a dropdown menu labeled 'Badurls Table'. Below it, the text 'BADURL@VIEW' is displayed. The main area contains a table with four columns: 'id', 'url', 'time', and 'status'. The table lists five entries, all with a status of 'BAD'. Below the table, there are search and sorting options. The search section includes three checkboxes: 'Search by id', 'Search by Url', and 'Search by Time', followed by a search input field and a 'Search' button. The sorting section includes a 'Clear Pane' button, an 'Order by column selection' button, and three checkboxes: 'Order by id', 'Order by Url', and 'Order by Time'. A note states: 'First click is DESCENDING , next is ASCENDING and so forth'.

id	url	time	status
2	Example.com	2021-12-18 ...	BAD
7	Example.com	2021-12-19 ...	BAD
9	Example.com	2021-12-19 ...	BAD
15	Exampler.c...	2021-12-19 ...	BAD
17	Exampler.c...	2021-12-19 ...	BAD

Tabelul detine informatii despre acesarea unor adrese ce sunt “ilegale”(Nu exista, sau am primit erori de tipul 40x, 301 , sau nu avem access de loc)

## Vederea a tabelului cu Rezultate

News Spider Database Manager

Results Table

RESULTS@VIEW

id	url	time	result	useragent
0	Example...	2021-12-...	wwwwww	Mozilla/5...
1	Example...	2021-12-...	qweqweq	Mozilla/5...
2	Example...	2021-12-...	Nothing f...	Mozilla/5...
3	Example...	2021-12-...	wwwwww	Mozilla/5...
5	Example...	2021-12-...	qweqweq	Mozilla/5...
7	Example...	2021-12-...	Nothing f...	Mozilla/5...
9	Example...	2021-12-...	Nothing f...	Mozilla/5...
11	Exampler...	2021-12-...	wwwwww	Mozilla/5...
13	Exampler...	2021-12-...	qweqweq	Mozilla/5...
15	Exampler...	2021-12-...	Nothing f...	Mozilla/5...
17	Exampler...	2021-12-...	Nothing f...	Mozilla/5...
18	https://th...	2021-12-...	Dire end ...	Mozilla/5...
19	https://th...	2021-12-...	Dire end ...	Mozilla/5...
20	https://th...	2021-12-...	US Bide...	Mozilla/5...
21	https://th...	2021-12-...	US Bide...	Mozilla/5...
38	https://w...	2021-12-...	<img alt=...	Mozilla/5...

☐ Search by id   
 ☐ Search by Url   
 ☐ Search by Time   
 ☐ Search by Result   
 ☐ Search by UserAgent

Search

Clear Pane    Order by column selection

First click is DESCENDING , next is ASCENDIND and so forth

☐ Order by id   
 ☐ Order by Url   
 ☐ Order by Time   
 ☐ Order by Result   
 ☐ Order by UserAgent

Tabelul detine informatii despre fiecare rezultat gasit sau asociat unui keyword la:

-data respective(inclusive milisecunda)

-user-agentul folosit

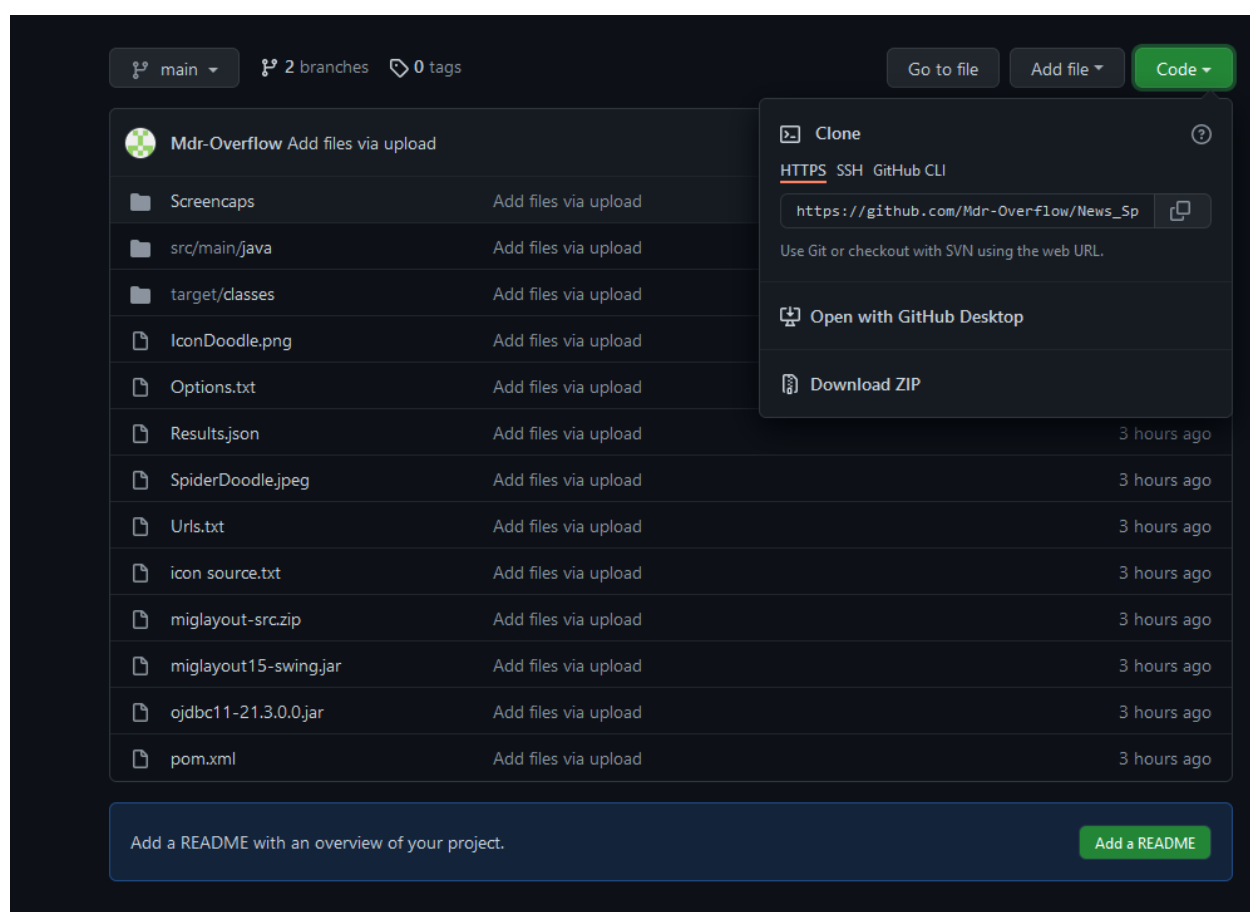
-Adresa Url

## Manual de Utilizare

Codul , cat si executabilul asociat aplicatiei se gaseste pe github prin utilizarea linkului: [https://github.com/Mdr-Overflow/News\\_Spider-WebScrapperProject](https://github.com/Mdr-Overflow/News_Spider-WebScrapperProject)

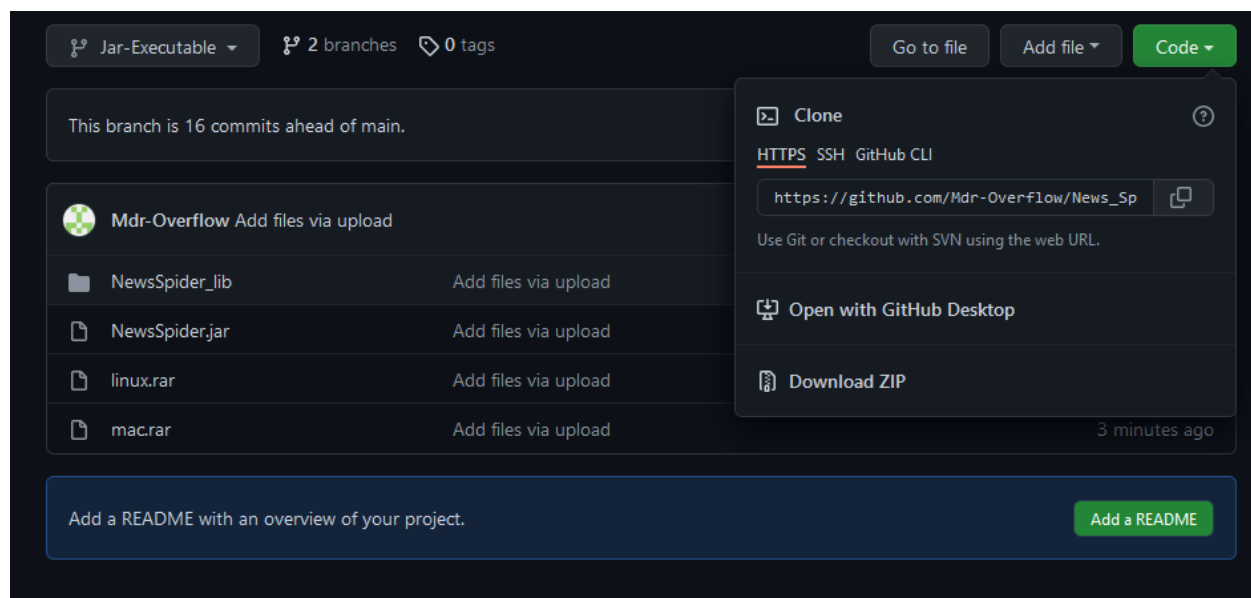
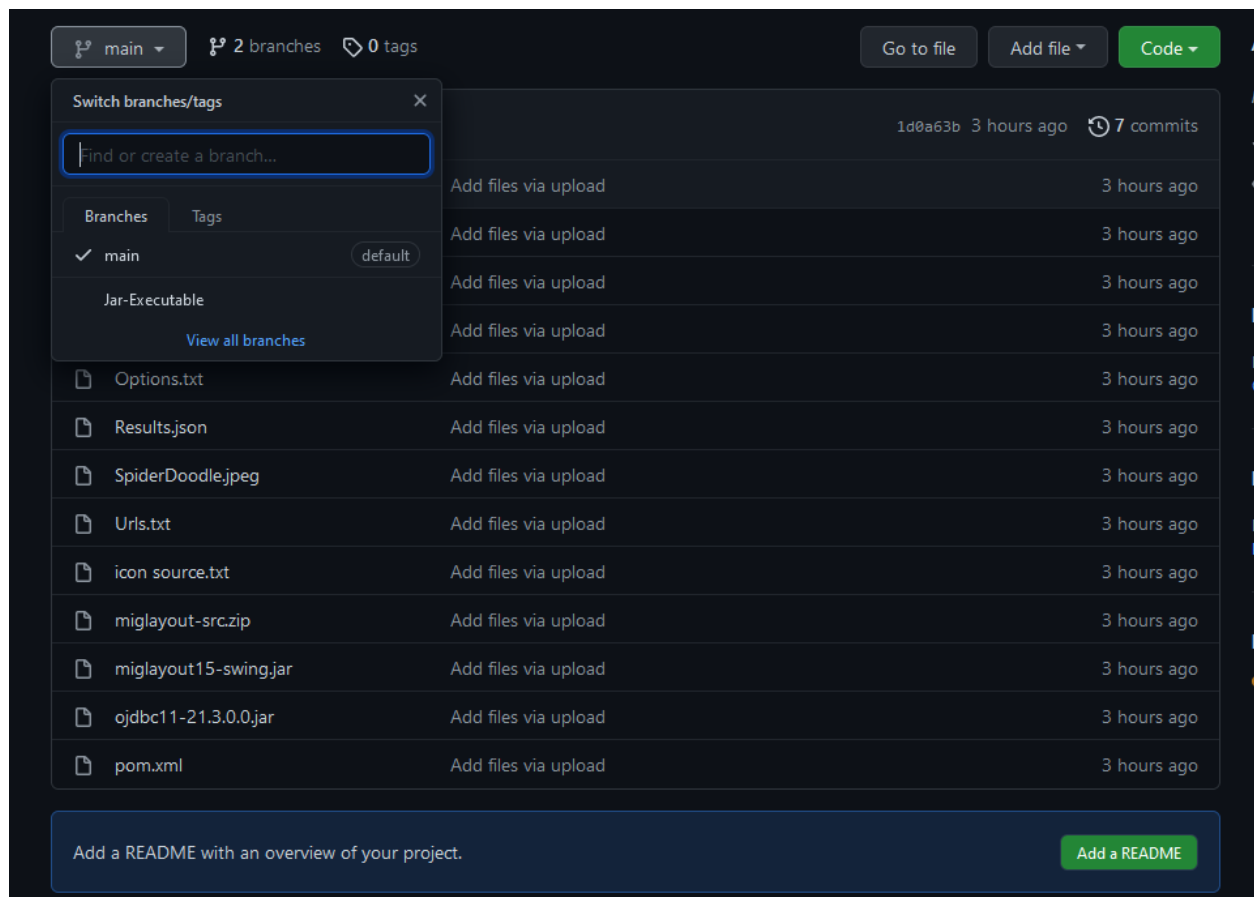
Odata ajunsi pe acesata adresa se poate descarca codul sau executabilul.

Codul se poate descarca dand click pe “Download zip” si apoi se dezarhiveaza.

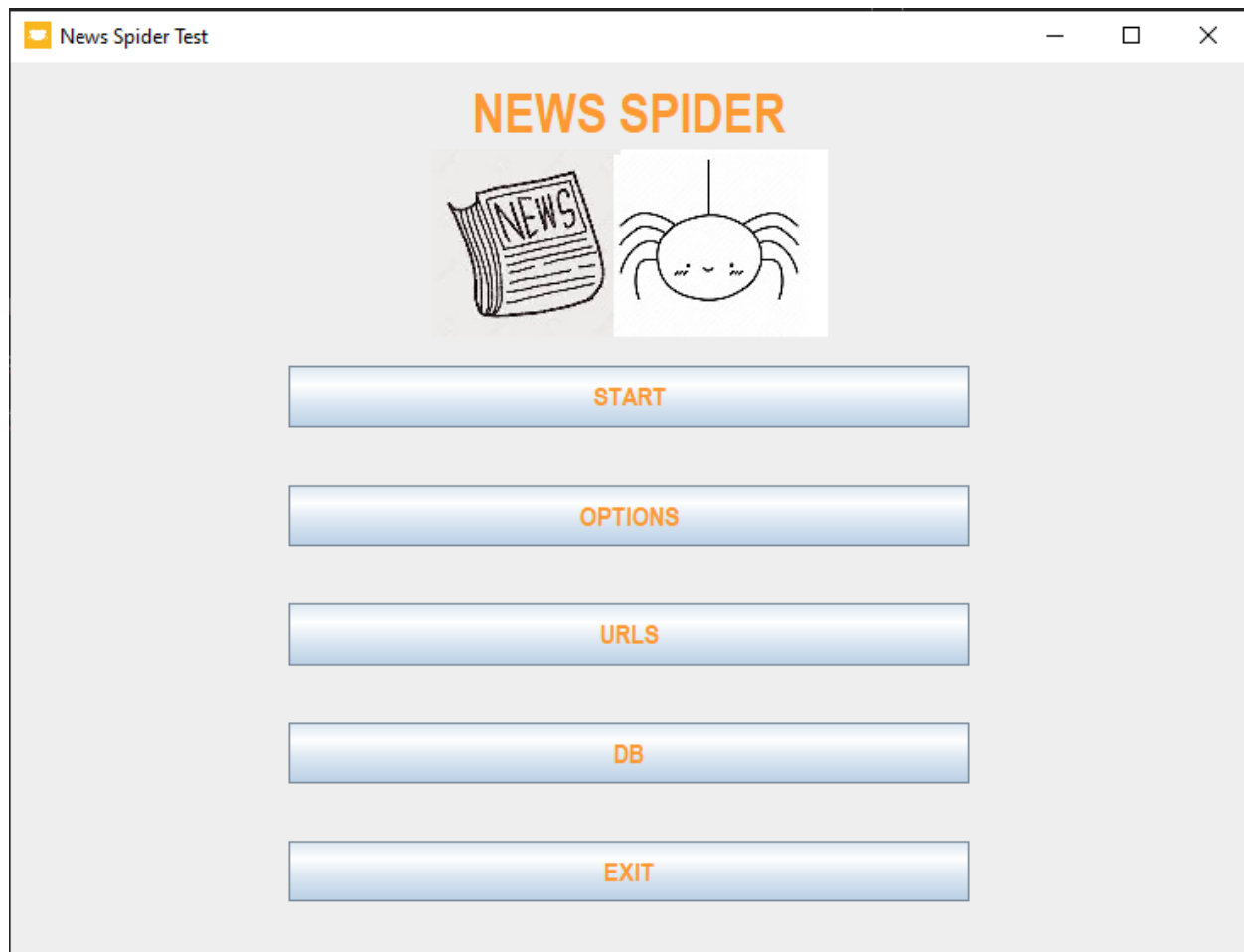


Executabilul se poate descarca dand click pe butonul de branch pe care scrie “main” iar apoi selectand “Jar-Executable”.

Dupa aceea se procedeaza ca si la cod.



Executabilul se va rula, odata ce este “depozitat” intr-un anumit director impreuna cu resursele necesare rularii.



In urma executarii fisierului .exe utilizatorul va fi prezentat cu un meniu care poate fi redimensionat, din care va putea alege 5 optiuni prin apasarea unui buton.

Prin selectarea butonului de exit se va inchide aplicatia(de asemenea prin selectarea “x”-ului specific oricarei aplicatii)

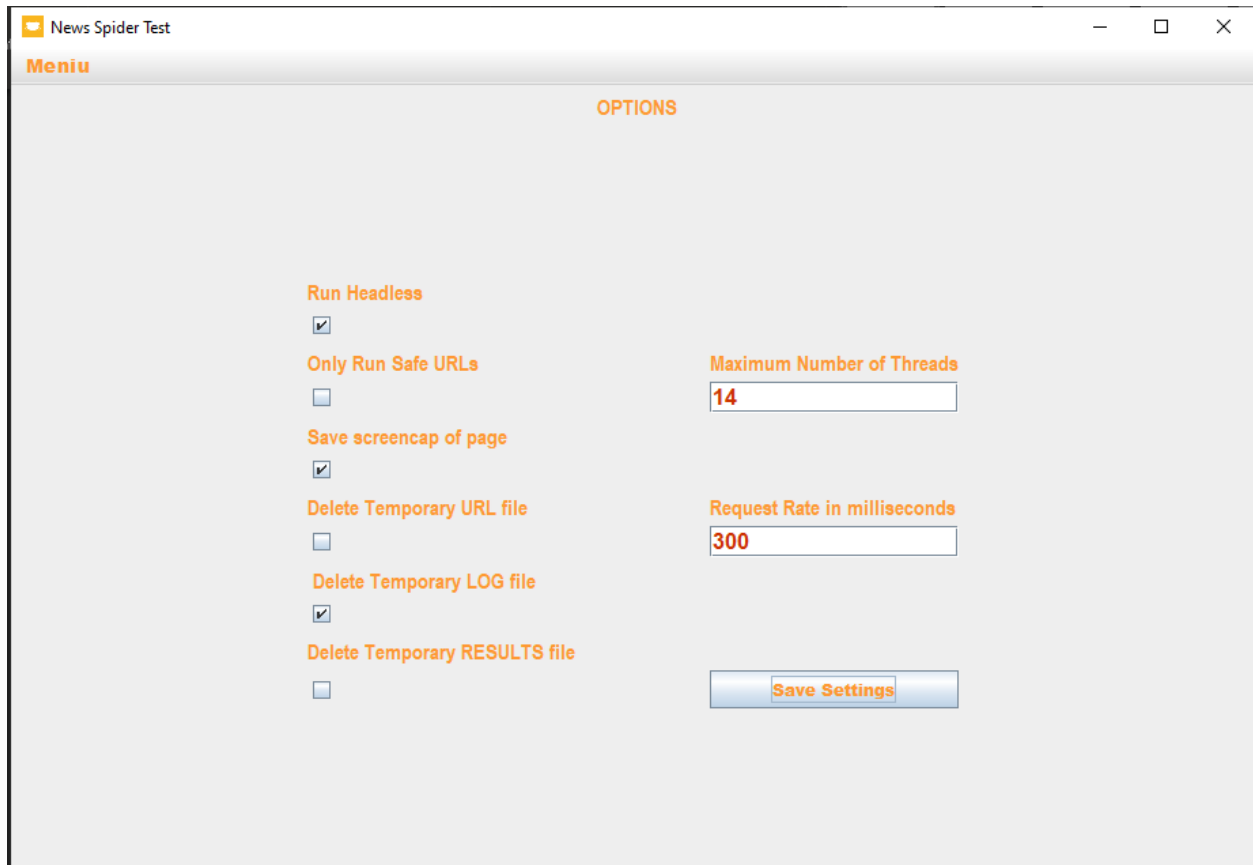


In urma exercitarii dorintei de “exit” din aplicatie , acesta va fi instiintat printr-un pop-up window daca doreste acest lucru.

Initial utilizatorul va trebui sa intre in meniul de optiuni pentru a schimba parametrii cautarii dupa dorinta sa , daca nu va dorii cei de “default”



## Valori de “Default”



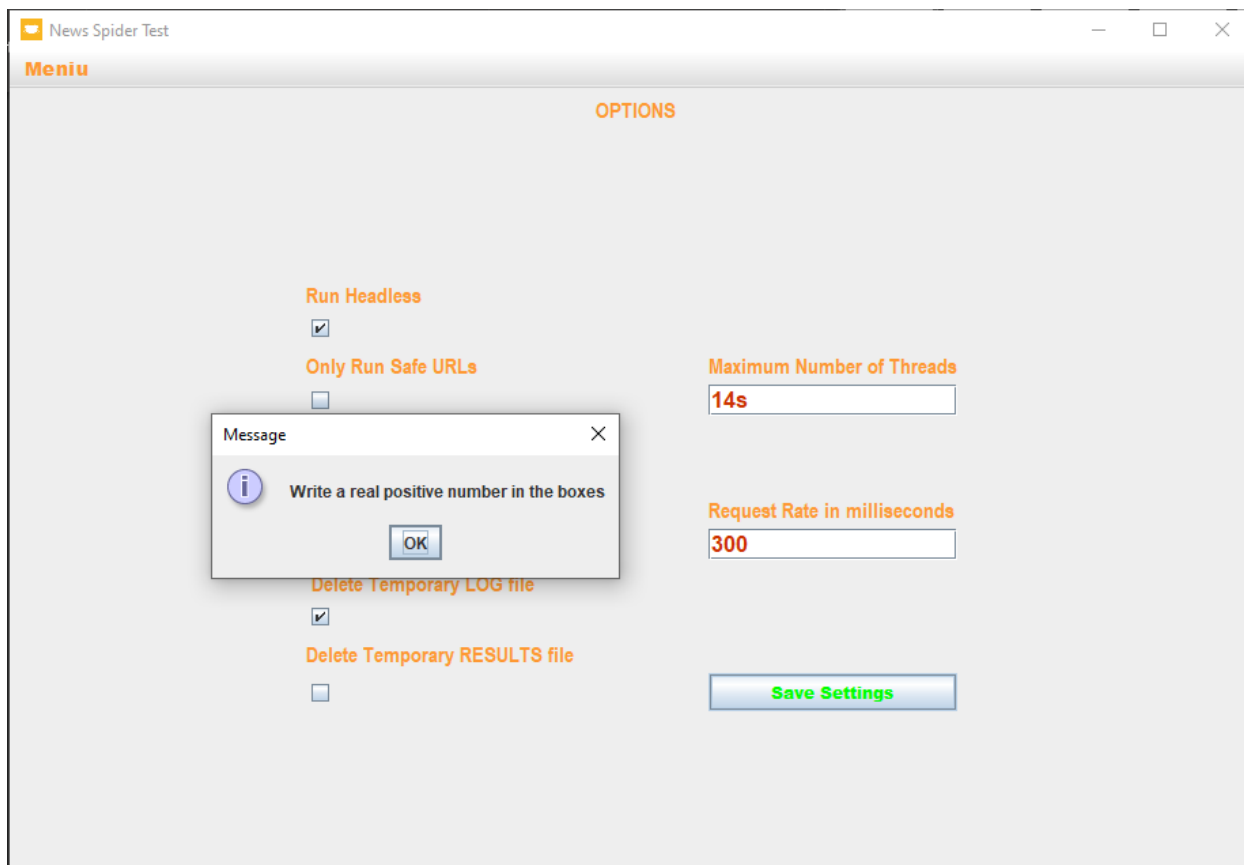
Prin selectarea unui dintre cele 6 checkbox-uri , utilizatorul va putea:

- vedea in “real-time” procesul de cautare(prin de-selectarea optiunii headless)
- rularea de adrese URL care nu se regasesc in baza de date cu adrese ilegale
- salvarea unui screenshot al paginilor parcurse cu succes
- stergera fişierelor temporare(adica fişierul de logging, de optiuni si rezultate.json)

Prin scrierea in cele 2 textbox-uri utilizatorul va putea alege numarul de fire de executie pe care va

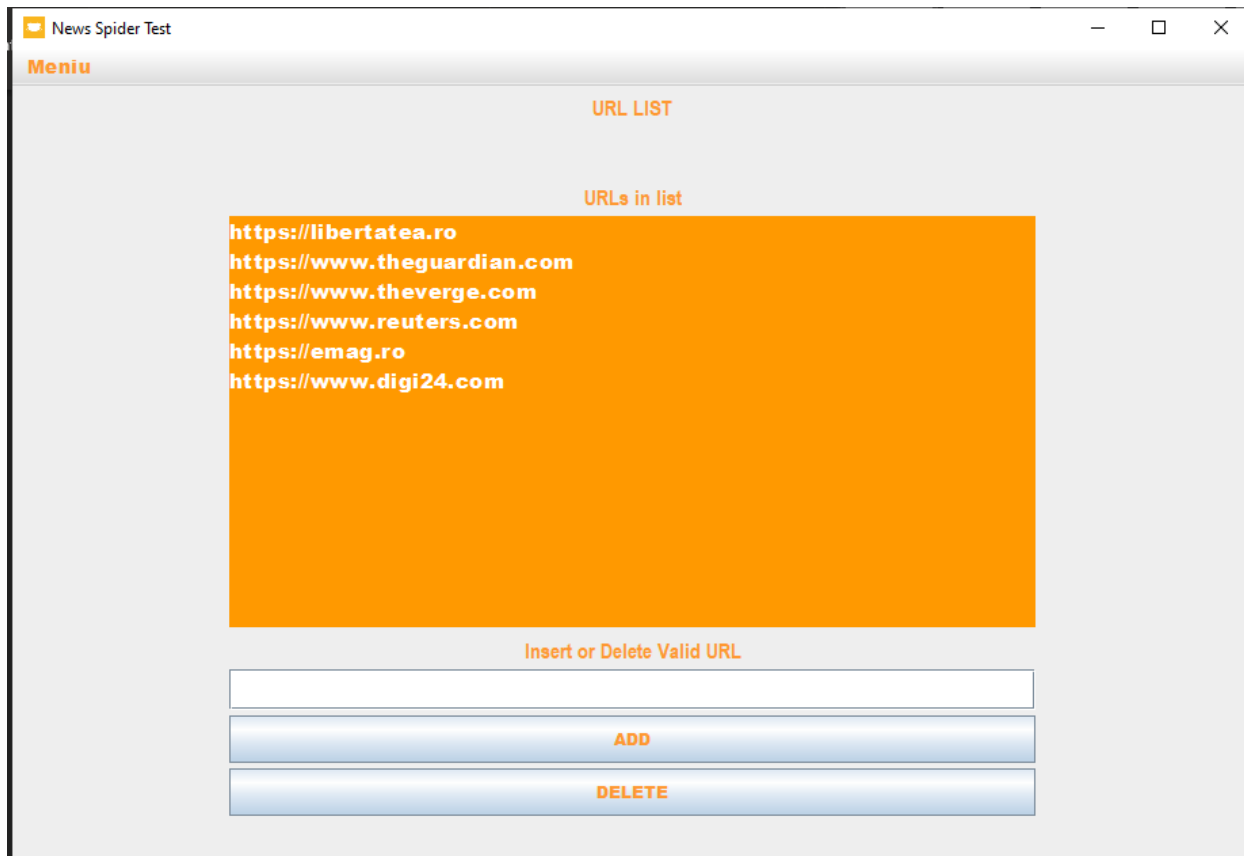
runa in backend cerinta de cautare, cat si rata de request asteptata

! Atentie, acestea vor fi salvate doar prin selectarea butonului de “Save Settings”



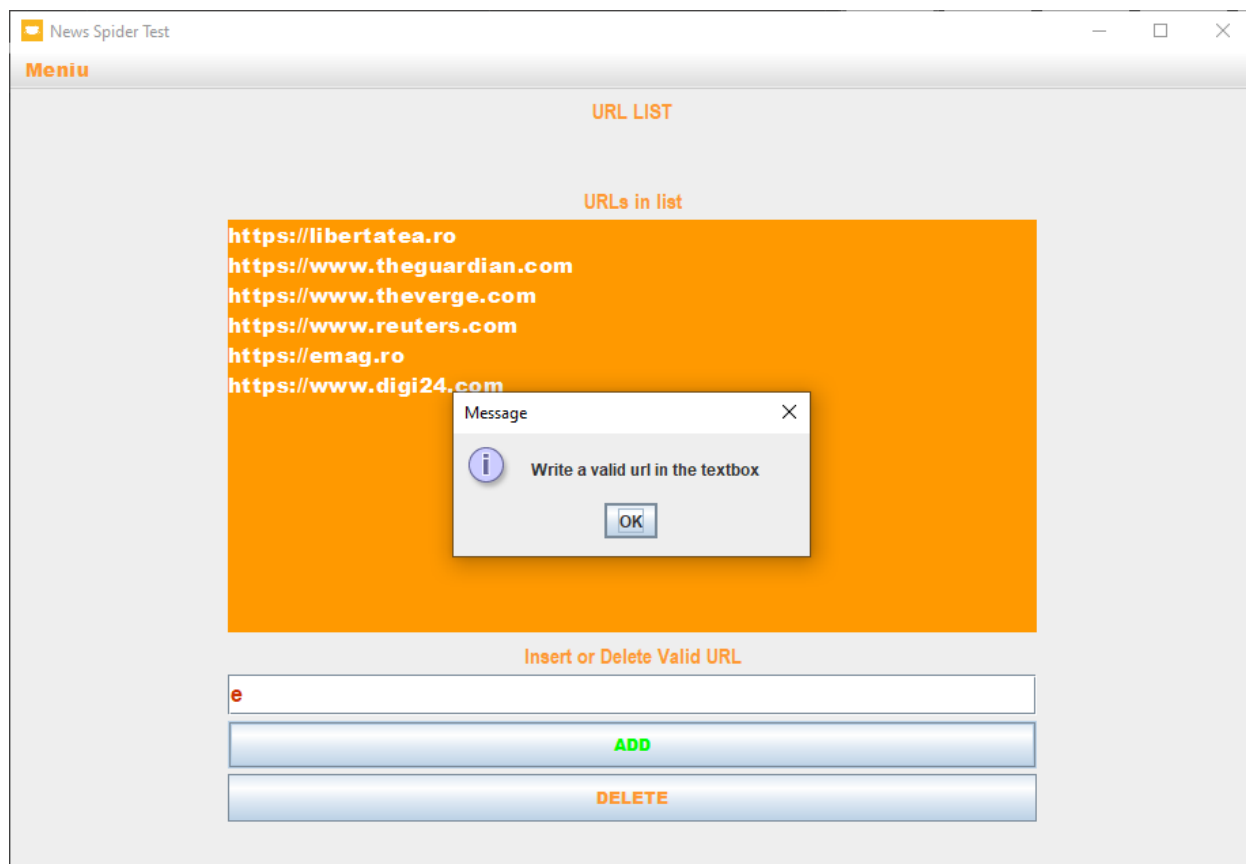
De asemenea utilizatorul va fi instiintat intru a insera valori posibile pentru aceste doua proprietati.

Utilizatorul este incurajat in a intra in meniul de Adrese URL pentru a insera sau sterge adrese pe langa cele date ca si default din care se vor extrage datele.



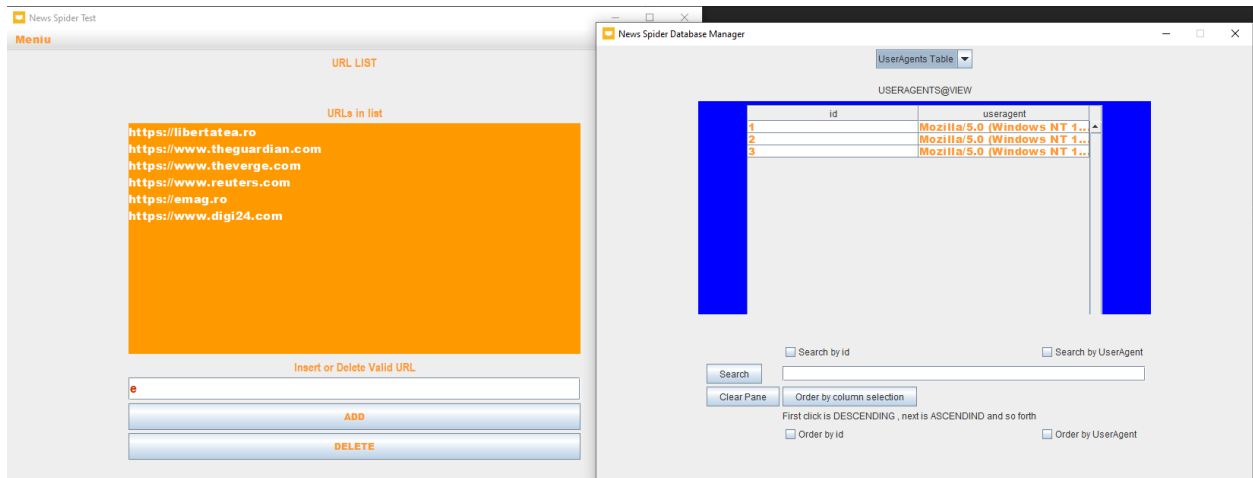
Prin selectarea butonului de “ADD” utilizatorul va putea adauga un nou string URL .

Prin apasarea butonului de “DELETE” utilizatorul va putea sterge un URL din lista, sau toate identice.

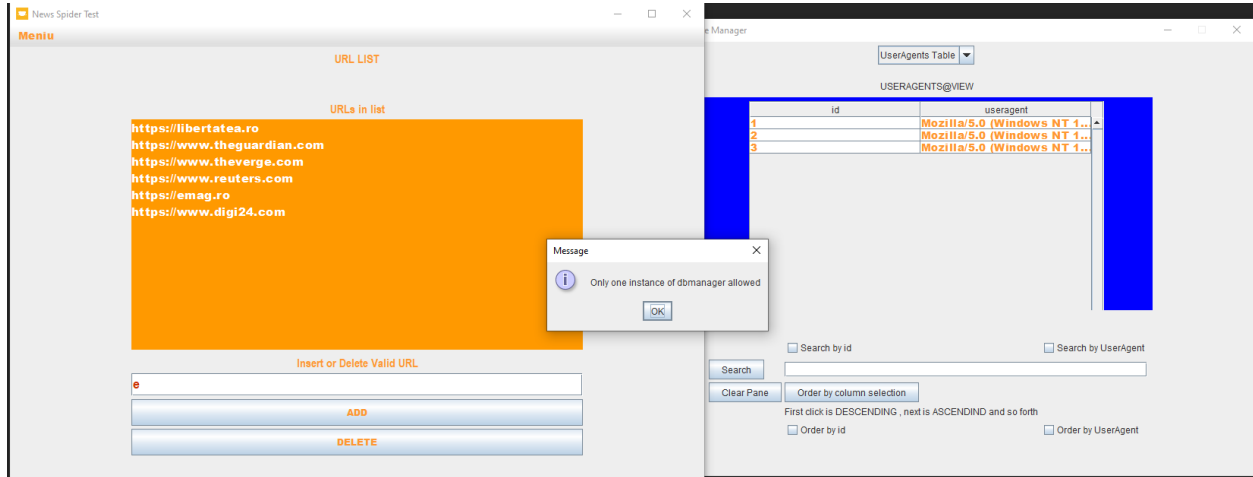


De asemenea utilizatorul este instiintat in vederea inserarii unei valori reale pentru o adresa URL.

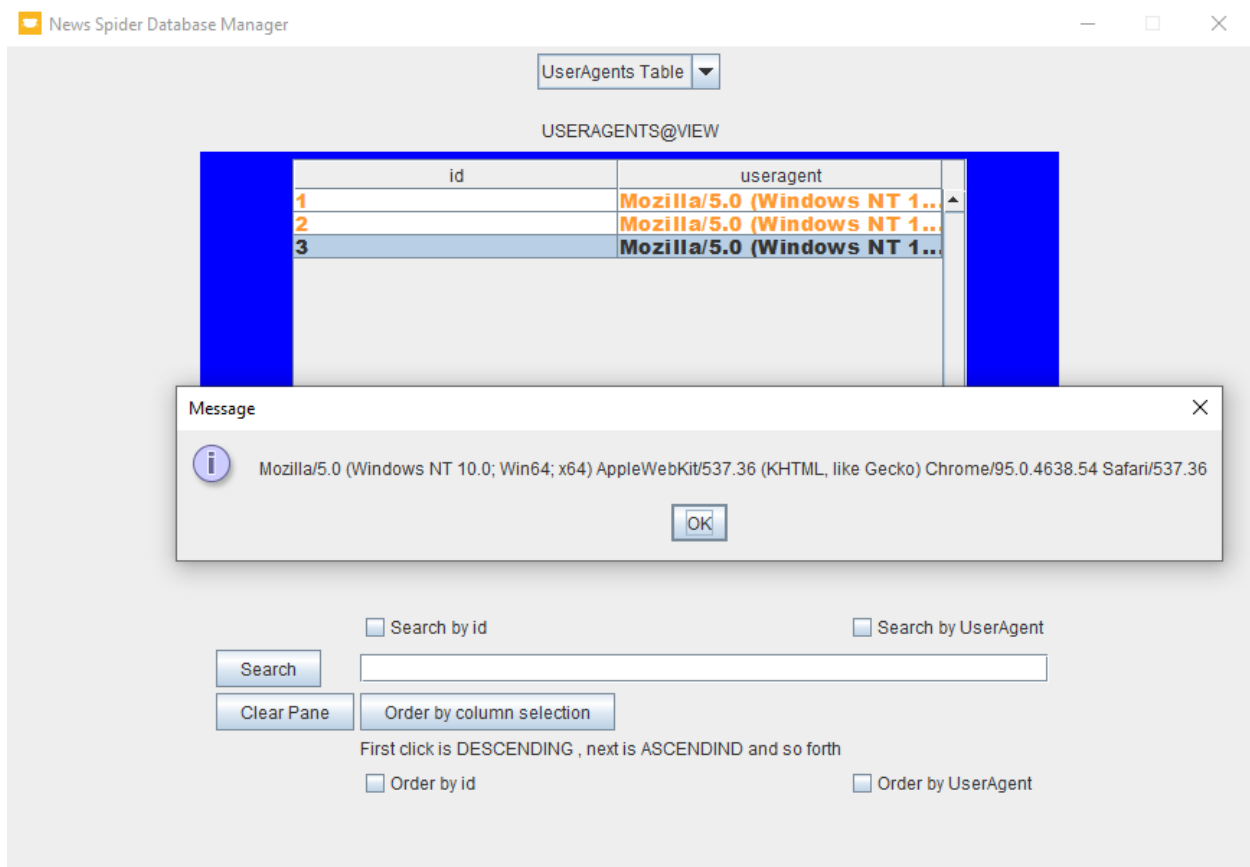
Utilizatorului ii este recomandat sa interactioneze si cu baza de date amintita anterior



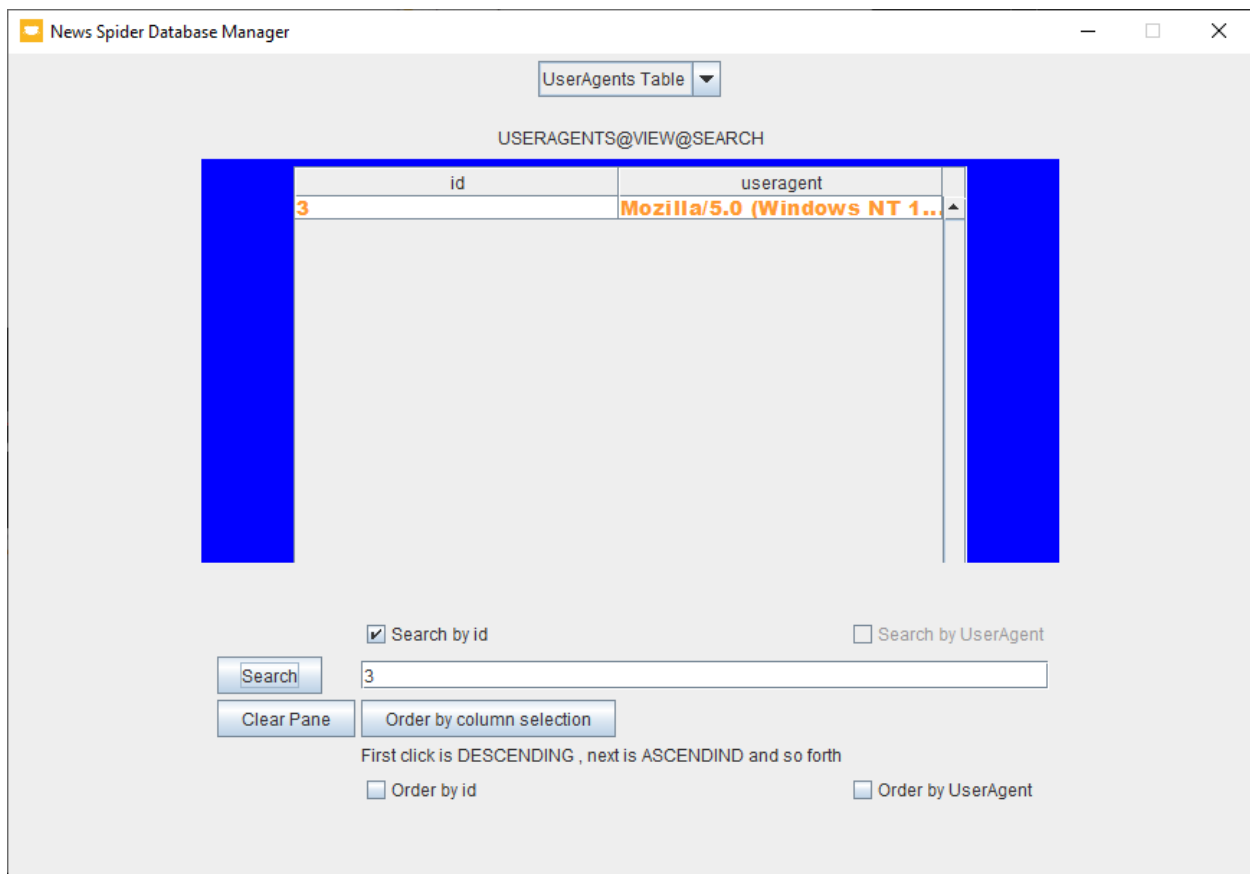
Ori prin selectarea butonului DB din meniul drop down, ori prin selectarea aceleiasi obtiuni din meniul principal utilizatorul va putea interactiona cu un “DB manager” unde va putea cauta anumite date.



De asemenea utilizatorul nu va putea deschide mai multe instante de “DB manager” , acesta fiind instiintat de acest lucru.



Utilizatorul va putea “vedea mai bine” datele specific prin efectuarea unui click asupra celulei din tabel aferente.



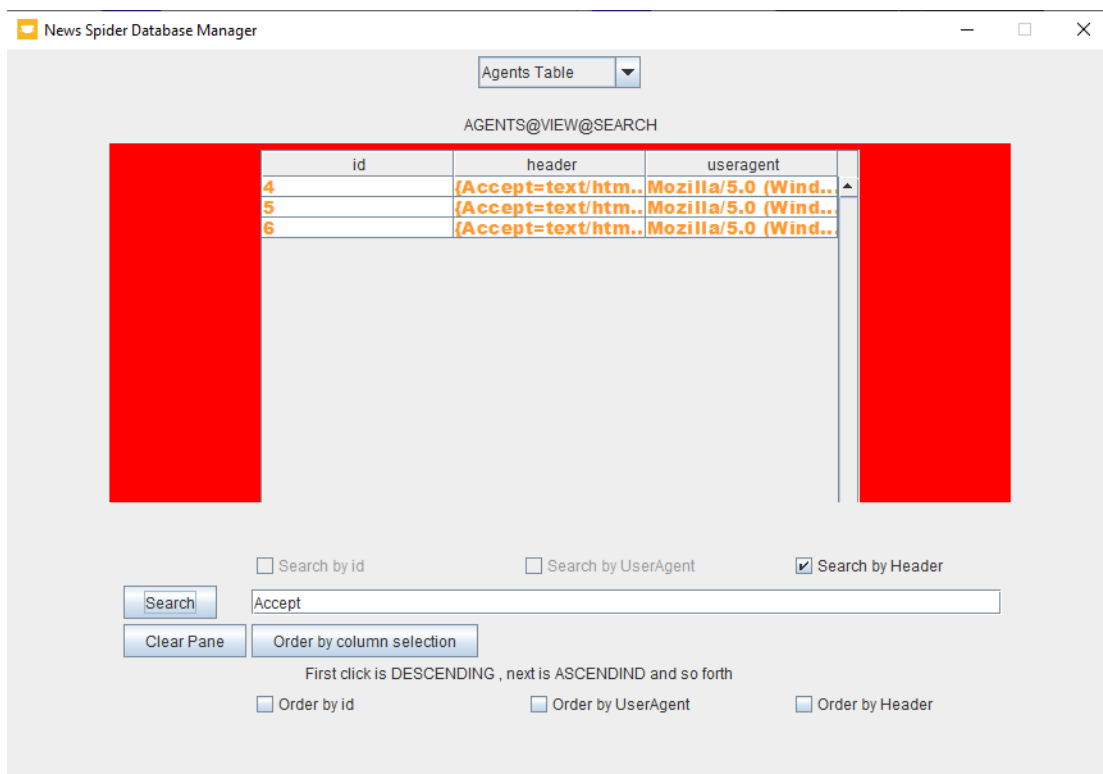
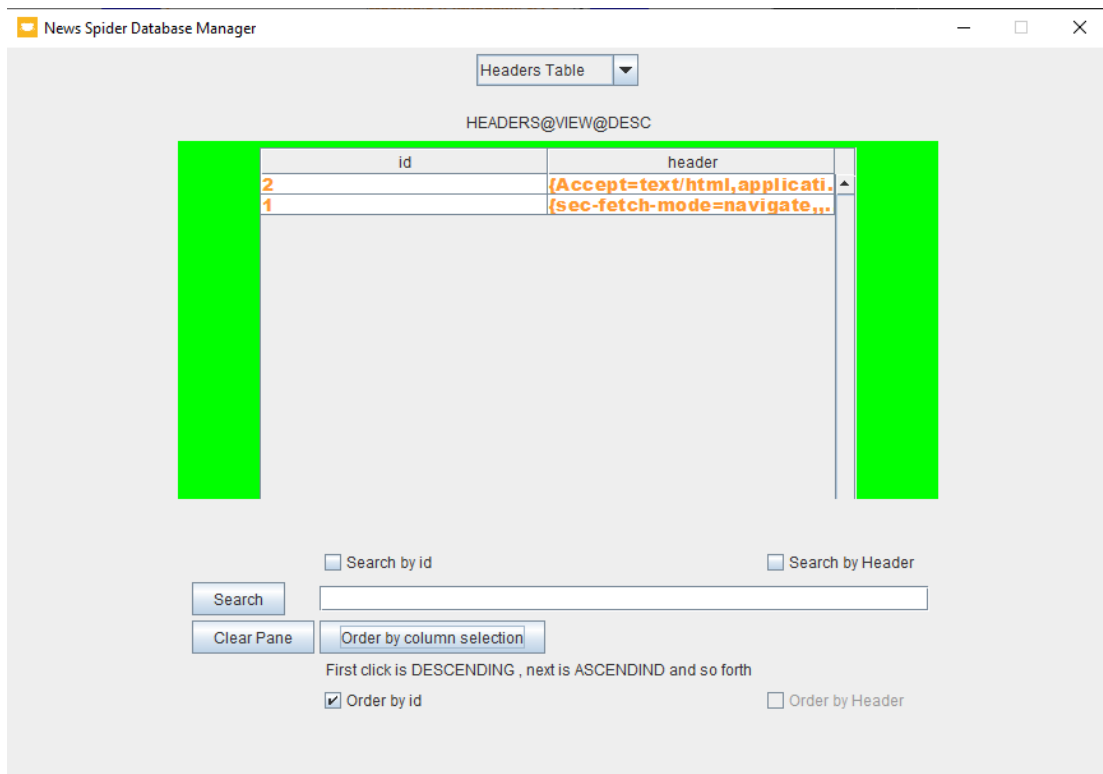
Utilizatorul va putea filtra datele prin apasarea butonului "Search" si alegerea coloanei aferente cautarii

De asemenea prin apasarea butonului "Order by column selection" si alegerea coloanei aferente , se va putea ordona descrescator elementele tabelului , pt a le ordona crescator se va apasa incodata butonul,

Prin apasarea butonului "Clear Pane" se va revenii la starea initiala a tabloului.

Aceasta functionalitate este regasita si pentru celealte tabele la care se va putea naviga selectand o optiune din option-selectorul din susul paginii.

Exemple:





News Spider Database Manager

Results Table

RESULTS@VIEW

id	url	time	result	useragent
56	https://w...	2021-12-...	...	Mozilla/5...
57	https://w...	2021-12-...	Biden sa...	Mozilla/5...
58	https://w...	2021-12-...	Boeing a...	Mozilla/5...
59	https://w...	2021-12-...	Boeing a...	Mozilla/5...
60	https://w...	2021-12-...	Dire end ...	Mozilla/5...
61	https://w...	2021-12-...	Dire end ...	Mozilla/5...
62	https://w...	2021-12-...	US Bide...	Mozilla/5...
63	https://w...	2021-12-...	US Bide...	Mozilla/5...
64	https://ti...	2021-12-...	Biden Tri...	Mozilla/5...
65	https://ti...	2021-12-...	How Bid...	Mozilla/5...
66	https://ti...	2021-12-...	Sinking a...	Mozilla/5...
67	https://ti...	2021-12-...	Biden's P...	Mozilla/5...
68	https://ti...	2021-12-...	It was ex...	Mozilla/5...

Message

Sinking approval ratings coupled with rising inflation may have weakened Biden's negotiating position on the Build Back Better bill

OK

Search

Clear Pane

Order by column selection

First click is DESCENDING , next is ASCENDING and so forth

☐ Order by id
 ☐ Order by Url
 ☐ Order by Time
 ☐ Order by Result
 ☐ Order by UserAgent

News Spider Database Manager

Badurls Table

BADURL@VIEW

id	url	time	status
2	Example.com	2021-12-18 ...	BAD
7	Example.com	2021-12-19 ...	BAD
9	Example.com	2021-12-19 ...	BAD
15	Exampler.c...	2021-12-19 ...	BAD
17	Exampler.c...	2021-12-19 ...	BAD

Message

2021-12-19 00:40:39.263

OK

☐ Search by id
 ☐ Search by Url
 ☐ Search by Time

Search

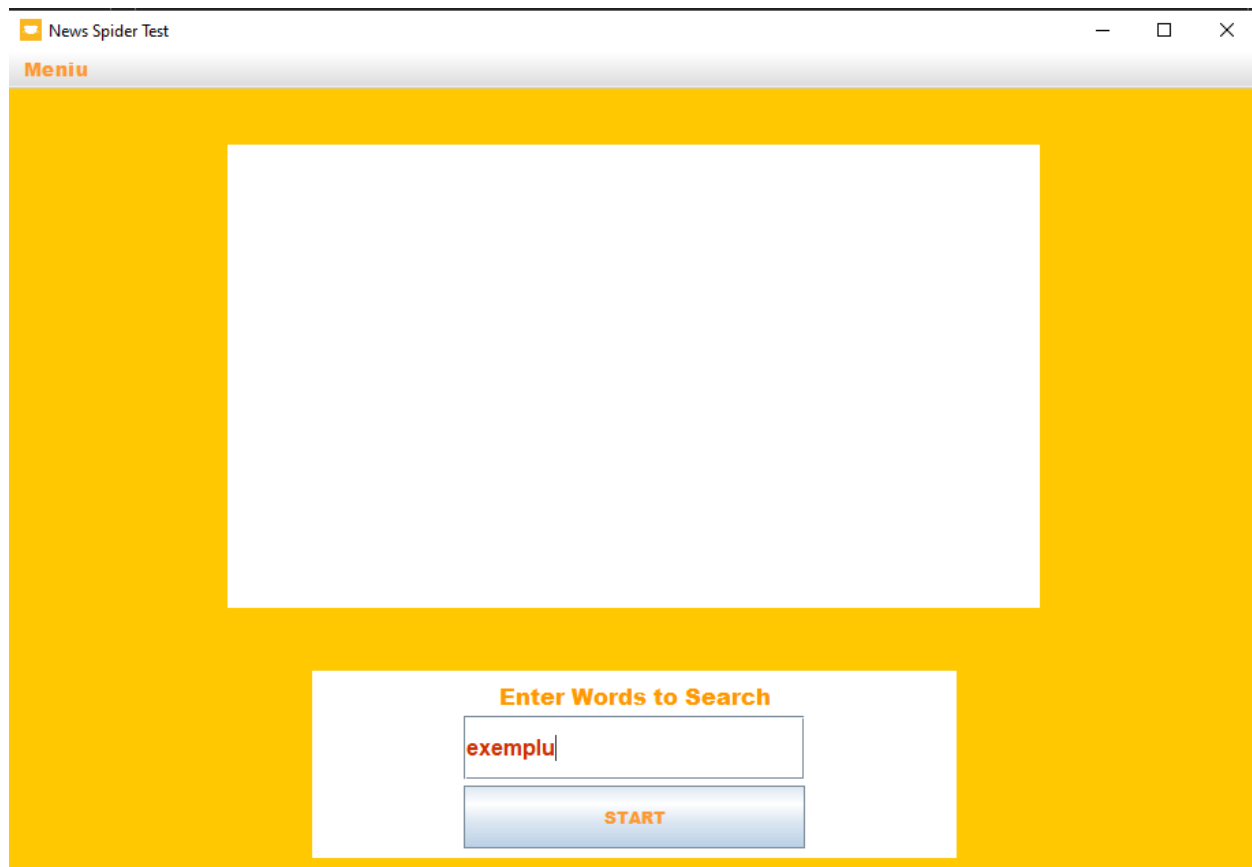
Clear Pane

Order by column selection

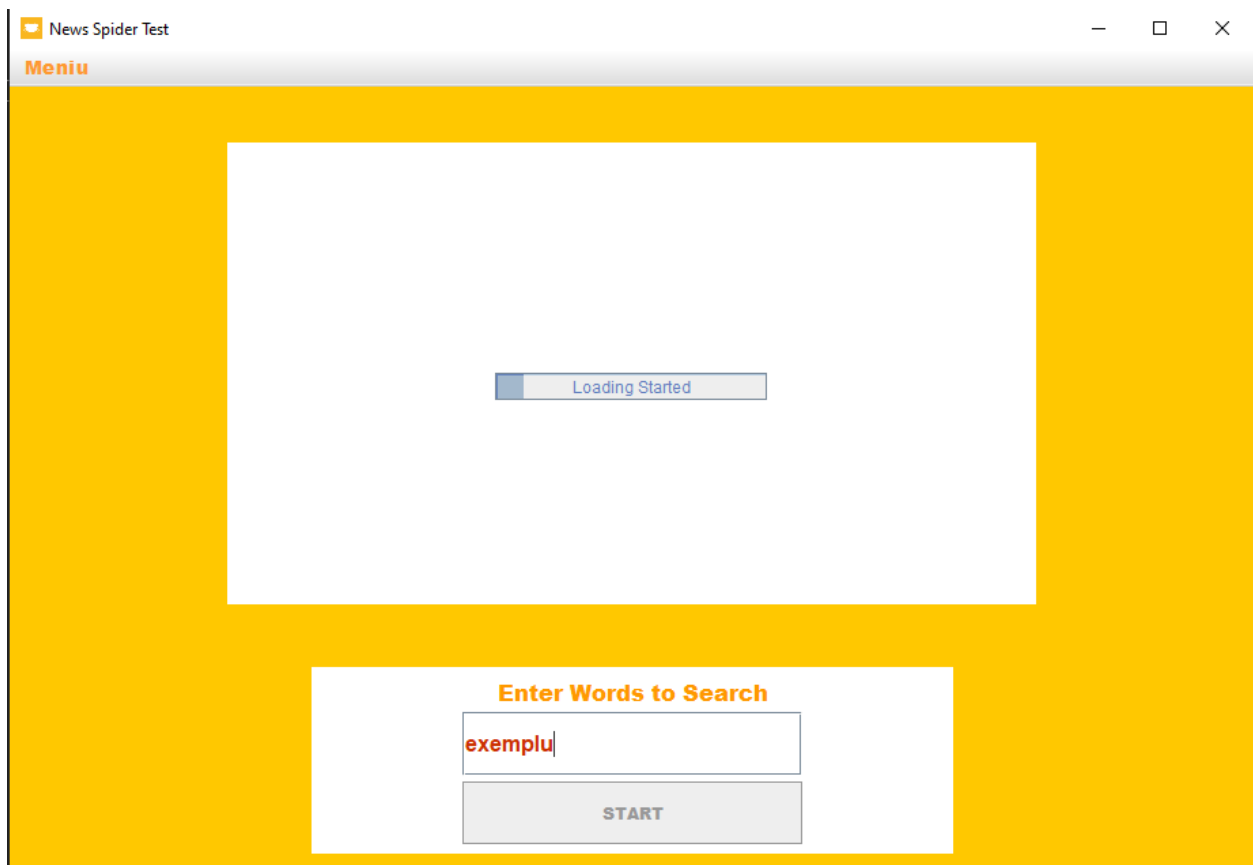
First click is DESCENDING , next is ASCENDING and so forth

☐ Order by id
 ☐ Order by Url
 ☐ Order by Time

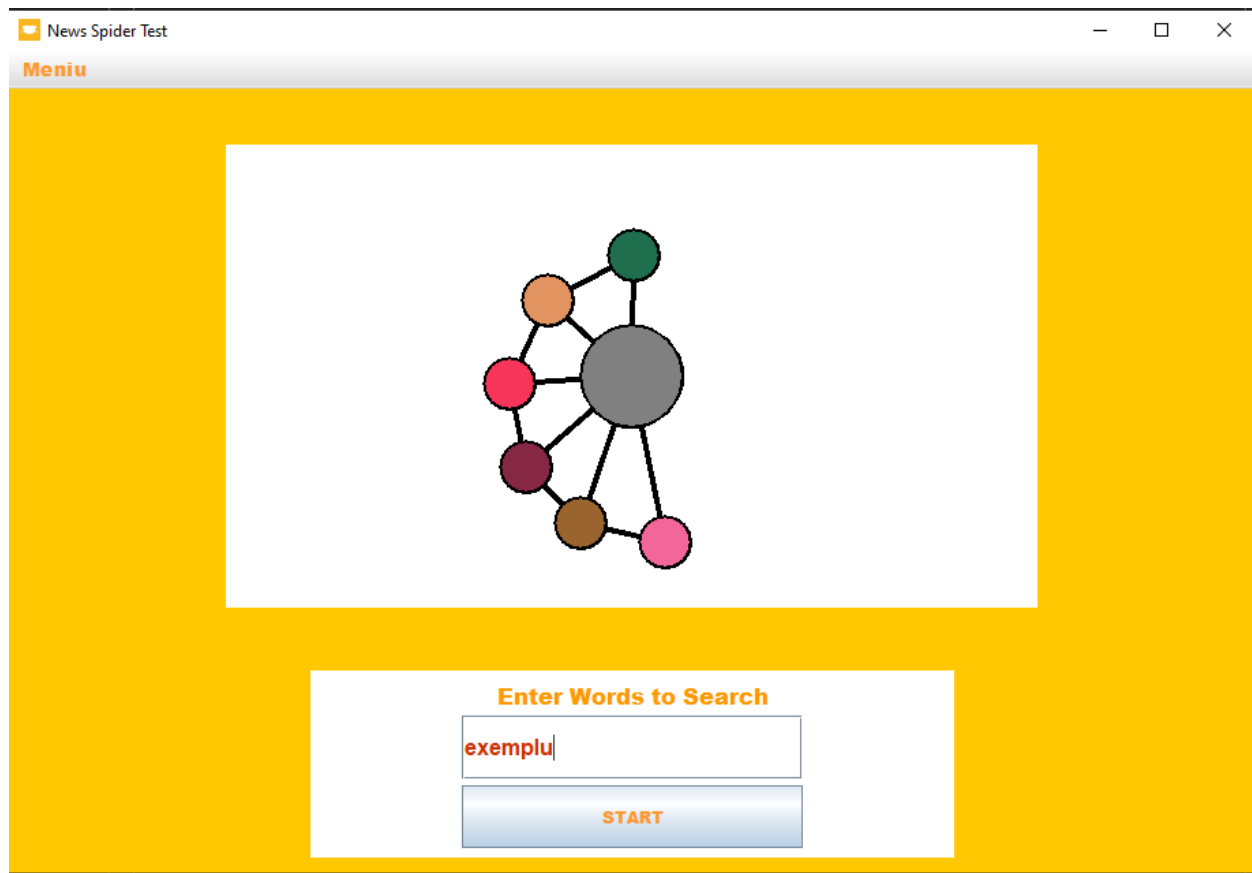
Utilizatorul va putea incepe cautarea prin selectarea butonului APP din drop down meniu, sau Start din main meniu.



Utilizatorul va putea scrie cuvatul cautat ("de gasit") dorit in textboxul din josul paginii, iar cautarea va incepe prin apasarea butonului de Start.



In timp ce va avea loc cautarea, utilizatorul se va putea orienta dupa bara de loading , si va putea face orice altceva in acelasi timp(cautarea are loc in Background intr-un thread separate)



Utilizatorul va putea acum vedea rezultatele prin selectarea unui din nodurile din graf, sau se va putea uita in baza de date la tabela rezultate.

## Capitolul 4.

### Probleme si Limitari

Bug-uri cunoscute: Parser-ul preia si text din DOM-ul paginii , astfel ca pot exista elemente de css/ html / javascript / jquery/ etc ce apar la textul extras.(Rezolvata)

Request-uri catre pagini care nu exista, dar au structura valida (de exemplu: <https://exemplu.com> ) suspenda thread-ul pe care ruleaza insa nu inchide browser-ul.(Rezolvata)

In unele size-uri ale paginii nodurile nu pot fi actionate din locul und e ar trebuii sa fie( functia afina care mapeaza coordonatele nodurilor pictate pe panel nu mapeaza corect coordonatele componentelor invizibile din spatele lor la care este facut handle).(Partial rezolvata)

Limitari:

- “Real world solutions” folosesc metode de a eluda detectia mult mai eficiente pe care nu le pot obtine(Ip pool rotativ de proxy-uri rezidentiale),
- Deoarece nu exista un standard pentru pop-up ul de gdpr sau acceptare de cookie-uri (unele site-uri le pun intr-un frame care il adauga apoi la DOM printr-un event, altele il pun pur si simplu intr-un <div> undeva in pagina , altele ofera un frame cu multe alte butoane cu text sau metadata asemanatoare) aplicatia nu va putea parsa unele pagini in implementarea actuala
- Multe pagini web ruleaza foarte mult jquery/javascript/etc. si dureaza mult pana cand instanta de browser incarca pagina , astfel thread-ul ce tine executia instantei de browser i-si poate oprii executia dupa mult timp(>2 s ) ceea ce scade mult din performanta de timp a aplicatiei

## Capitolul 5.

### Sumar al lucrarii

Proiectul este realizat in proportie de 100% , avand in cosiderare functionalitatile promise in contextul initial al proiectului(functionalitati non-aspirationale), cat si observatiile relatate in versiunea beta.

Initial, versiunea Beta a proiectului a fost dezvoltata in Python, insa am decis sa rescriu proiectul in Java, avand in vedere asemanarea unor lib/frameworkuri (Threading,Playwright), cat si a posibilitatii de scalabilitate a aplicatiei.

Dintre functionalitatile initial denumite “aspirationale” nu au fost implemenate idei precum “IP Pool rotativ” sau parsare avansata , avand in vedere compexitatea sau lipsa de resurse(Proxy generator , sau lista de proxy-uri ), graful UML initial al functionalitatii din spatele scraper-ului denotand acest lucru (lipsa “Proxy checkerului” sau a unei DB sau generator de proxy-uri, colorate cu rosu).

Functionalitatea de web-crawler poate sa fie implementat avand in vedere existenta la nivel de backend a codului cat si a posibilitatii de “scalabilitate”.

Multe posibile bug-uri au fost corectate, insa acest lucru nu implica ne-existenta altora pe care nu le-am putut regasi in testare.

Aplicatia detine limitari, redate anterior, insa acestea pot fi rezolvate intr-o anumita masura avand in vedere posibilitatea de scalare a aplicatiei, dar necesita exponential mai mult timp.

## Capitolul 6:

### Viitoare Implementari/Directii Viitoare

Avand in vedere acentul pus pe “scalabilitate” in GUI si Backend, preconizat si in capitolele anterioare, viitoare implementari pot tine de:

- 1) Cresterea eficientei de timp a cautarilor de text, acest lucru poate fi realizata intr-o anumita proportie prin salvarea cookie-lor intr-un format binar intr-un “glob file” ce va fi retinut de baza de date, in acest mod nu va mai trebuii sa astept/simulez actiuni pe pagini(ceea ce va eficientiza de zeci de ori procesul) , in schimb folosind doar requesturi din jsoup pt. a prelua structura si textul paginii, request in care voi incarca cookie-urile preluate din baza de date pt. adresa respectiva si intr-un interval de timp valid(la fiecare 10 min se vor pornii requesturi prin intermediul Agent-Playwright precum este acum pt. a salva o schimbare de cookie-uri/politica a domeniului website-ului).
- 2) Metode de evitare a detectitei mai bune cat si de “trecere” de “bariere” precum Pop-upuri de gdpr, Captcha, etc.
- 3) Parsare mai complexa a paginii(utilizand termeni “asociati” textului cautat, produsi prin stocarea unor cuvinte regasite in fraza ce continue textul cautat, sau in elemente apropiate in DOM-ul paginii)(Dupa extragerea tuturor elementelor ce contin acel text, se va putea parsa prin “sentiment analysis” , pentru a reda proprietati precum rata de pozitivitate sau asemanare cu alte rezultate.)

## Grafic de Realizare al Proiectului :

ID	Data	Obiectiv	Descriere	Status	Obs.
1	01.10.2021	Definirea temei	Identificarea temei	Done	-
2	10.11.2021	Identificarea functionalităților	Schema UML Si gandire structura	Done	-
3	20.11.2021	Requesturi	Requesturi din Instanta de Playwright	Done	-
4	23.11.2021	MultiThreading	Pt. Instanetele de Playwright	Done	-
5	30.11.2021	- Vers. Beta -	Creearea de GUI  Si Multithreading pe  Frame-urile din GUI si procesele lor	Done	-
6	01.12.2021	Portarea In Java	Incepere Portarea In Java	Done	-
7	6.12.2021	Finalizare portare in java	Finalizare portare in java	Done	
8	8.12	GUI update	Implementarea de noi features  Ale GUI-ului	Done	
9	10.12	GUI final touches	Finalizare GUI	Done	
10	14.12	Baza de date	Creearea Bazei de Date ce var ula pe server din localhost sau altul dat	Done	



11	18.12	Baza de date finalizare	Finalizare aspect ce tin de baza de date  Si GUI pt. interactiunea cu baza de date	Done	
12	20.12	Bugfixing	Bugfixing si testare	Done	Nu au fost eliminate toate bug- urile posibile
13	22.12	Finalizare Proiect	Finalizare Proiect	Done	

## Referinte Bibliografice.

<https://www.zenrows.com/blog/mastering-web-scraping-in-python-from-zero-to-hero>

<https://playwright.dev/docs/intro>

<https://docs.oracle.com/javase/8/docs/api/index.html?javax/swing/package-summary.html>

<https://www.baeldung.com/jackson>

<https://docs.oracle.com/javase/7/docs/api/java/lang/Thread.html>

<https://www.baeldung.com/thread-pool-java-and-guava>

<https://www.baeldung.com/java-runnable-callable>

<https://stackoverflow.com>