

C++ Builder WebBrower

22.02.2021

Îndrumător:

dr. ing. Daniel Morariu

Student:

Bucur Dan-Alexandru (223/2)

Istoric versiuni

Data	Versiune	Descriere	Autor
30.10.2020	1.0	Crearea celor două forme a Browser-ului.	Bucur Dan-Alexandru
3.11.2020	1.1	Crearea unor toolbar-uri pentru o navigare mai ușoară a utilizatorului de asemenea implementarea unei componente TBrowser pentru navigarea pe internet și un Memo pentru afișarea conținutului fișierului citit.	Bucur Dan-Alexandru
17.11.2020	1.2	Legarea unor butoane de un searchbar alături de componenta TBrowser.	Bucur Dan-Alexandru
15.12.2020	1.3	Creare și implementarea clasei URL responsabila cu listele de URL-uri.	Bucur Dan-Alexandru
29.12.2020	1.4	Crearea și implementarea de metode pentru clasa URL.	Bucur Dan-Alexandru
11.01.2021	1.5	Crearea și implementarea clasei HTML și a moștenirii de către clasele I, B, BR și FINAL.	Bucur Dan-Alexandru
18.02.2021	1.6	Crearea și implementarea unui polimorfism necesar recunoașterii tagurilor.	Bucur Dan-Alexandru

Cuprins

Cuprins

Istoric versiuni.....	2
Cuprins.....	3
1 Specificarea cerințelor software.....	4
1.1 Introducere.....	4
1.1.1 Obiective.....	4
1.1.2 Definiții, Acronime și Abrevieri.....	4
1.1.3 Tehnologiile utilizate.....	5
1.2 Cerințe specifice.....	5
2 Funcționalitate.....	5
2.1 Citirea fișierelor HTML.....	5
2.1.1. Descriere.....	5
2.1.2. Flux de bază.....	5
2.1.3. Pre-condiții.....	5
2.1.4. Post-condiții.....	5
2.2 Navigarea Înapoi.....	5
2.2.1. Descriere.....	5
2.2.2. Flux de bază.....	6
2.2.3. Pre-condiții.....	6
2.2.4. Post-condiții.....	6
3 Implementare.....	6
3.1 Diagrama de clase.....	6
3.2 Descriere detaliată.....	7
4 Bibliografie.....	7

1 Specificarea cerințelor software

1.1 Introducere

Ne propunem dezvoltarea unui browser simplu capabil atât de navigarea pe internet cât și citirea fișierelor html.

1.1.1 Obiective

- Crearea unui browser care să poată naviga ușor înainte și înapoi prin lista de url-uri.
- Citirea și recunoașterea tagurilor html.
- Afișarea fișierelor citite într-un mod corespunzător.

1.1.2 Definiții, Acronime și Abrevieri

- head = pointer către primul element din listă, practic capul listei.
- curr = pointer către elementul curent din listă.
- temp = pointer către un element din listă cu uz temporal, în general a avut rolul de element anterior elementului curent(curr).
- url = membru al structurii LRU, de tipul AnsiString, el stochează url-ul propriu-zis.
- next = membru al structurii LRU, de tip pointer către un element din structură, el se ocupă cu parcurgerea listei și determinarea elementului curent.
- AddNode(AnsiString addData) = metodă a clasei URL, aceasta creează lista și o umple de elemente.
- DeleteNode() = metodă a clasei URL, aceasta șterge ultimul element adăugat în listă.
- AnsiString Last() = metodă a clasei URL, aceasta va returna ultimul element din listă.
- LRU = structură „încuibărită” în clasa URL.
- node = pointer de tipul structurii LRU.
- URL() = constructor implicit al clasei URL(asemenea și pentru clasele HTML, I, B, BR, FINAL).
- ~URL() = destructor al clasei URL(asemenea și pentru clasele HTML, I, B, BR, FINAL).
- File = membru public al clasei HTML, folosit la stocarea Ansistring-ului final.
- FilePath = membru public al clasei HTML, reține numele și extensia fișierului citit.
- i și j = membri publici ai clasei HTML, sunt contori.

- lineLength = membru public al clasei HTML, reține lungimea șirului.
- S[100] = tablou de membri publici ai clasei HTML, rețin AnsiString-uri.
- Temp = membru public al clasei HTML, reține un AnsiString curent.

1.1.3 Tehnologiile utilizate

- C++ Borland
- C++ Builder Embarcadero 10.3

1.2 Cerințe specifice

-Una din funcționalitățile specifice ale aplicației este evident căutarea pe internet după un URL dat de utilizator, stocat într-o listă a unui obiect.

-Posibilitatea navigării înapoi folosind lista de url-uri reprezintă de asemenea una din funcționalitățile specifice.

-Citirea și afișarea fișierelor HTML.

2 Funcționalitate

2.1 Citirea fișierelor HTML

2.1.1 Descriere

Am ales ca aceasta să fie prima funcționalitate deoarece este de fapt și cea mai importantă dintre ele. Din nefericire nu poate interpreta tag-urile pentru tabel ori scrierea cu bold ori italic efectivă.

Pe scurt se citește fișierul HTML dat, rând cu rând, iar apoi se afișează într-un memo.

2.1.2 Flux de bază

Deci primul lucru pe care utilizatorul îl notifică la rularea programului este faptul că nu poate decât să caute pe internet în prima formă, de aceea am adăugat un buton numit „HTML” care ascunde fereastra Browser și o afișează „html” unde se găsește butonul „Read” care face citirea și afișarea fișierelor html prin apel la metoda cu același nume Read() a clasei HTML (polimorfizată mai departe la clasele I, B, BR și FINAL).

2.1.3 Pre-condiții

Înainte de utilizarea funcției trebuie accesată forma a doua.

Interpretarea tagurilor bold și italic sunt posibile dar nu complet, deoarece nu se poate afișa textul corect, de asemenea tagurile pentru tabel nu pot fi interpretate ori citite deloc.

2.1.4 Post-condiții

După utilizarea funcției, trebuie să se revină la prima formă ca programul să fie închis corect.

2.2 Navigarea Înapoi

2.2.1 Descriere

Am ales această funcționalitate să o prezint deoarece este necesară oricărui browser.

Aceasta funcționează ca un motor de căutare simplist doar cu url-uri.

2.2.2 Flux de bază

Această funcționalitate este ușor de remarcat. O dată ce se apasă butonul „Go” începe căutarea site-ului corespunzător url-ului, în spate se crează o listă de url-uri care merge într-un singur sens. La apăsarea butonului „Back”, se șterge ultimul url adăugat și se navighează la penultimul url(penultimul devenind ultimul).

2.2.3 Pre-condiții

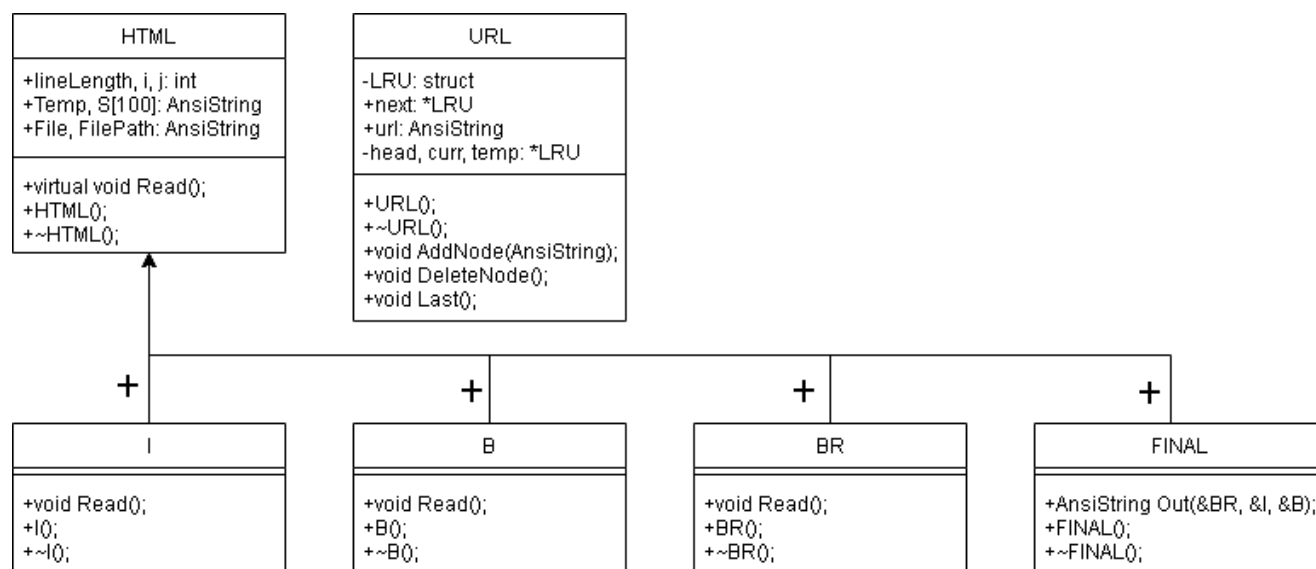
Înainte de utilizare trebuie obligatoriu apăsat butonul „Home” ca să se poată crea lista de url-uri corect.

2.2.4 Post-condiții

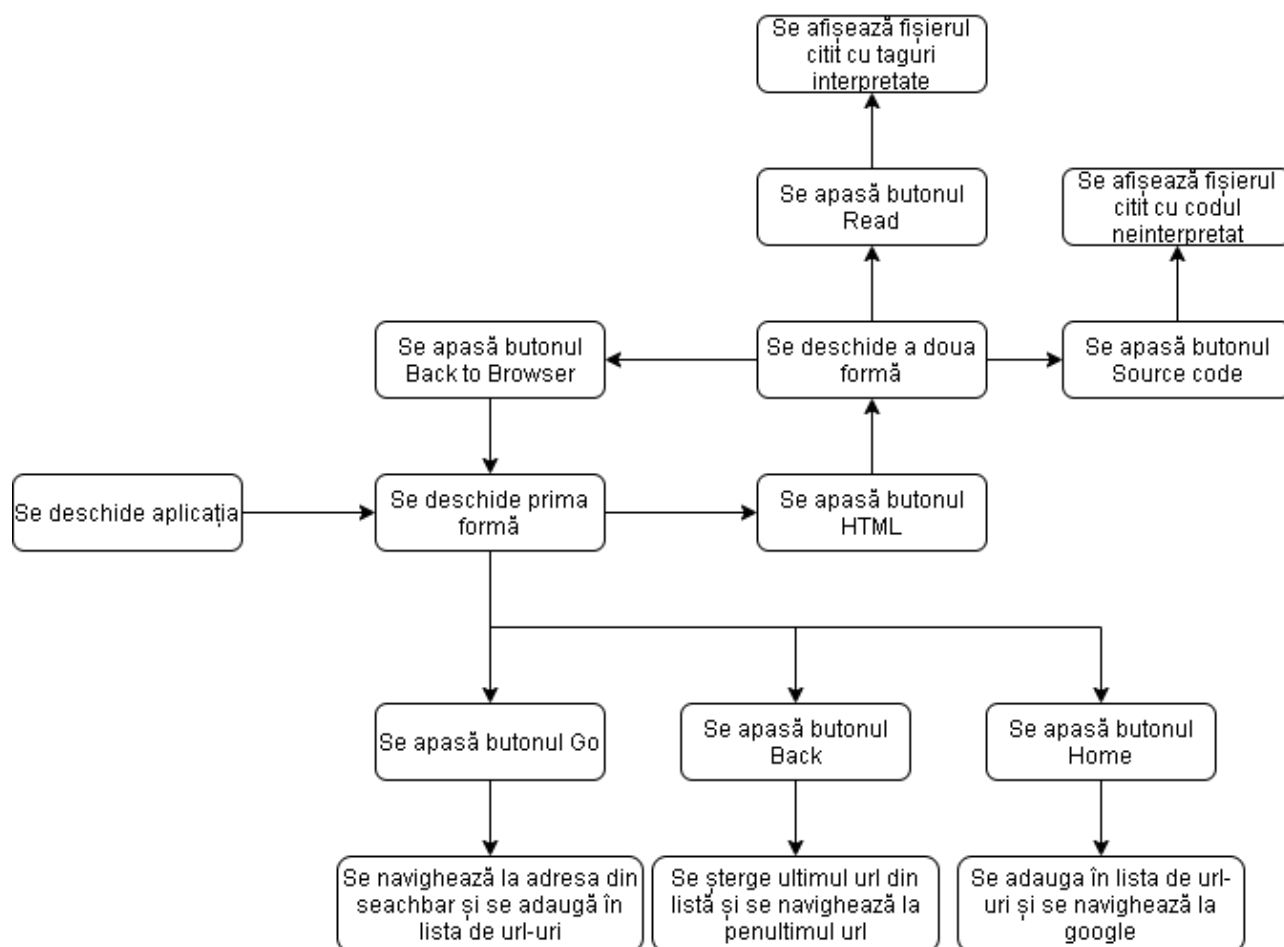
Dacă s-a apăsat de prea multe ori butonul „Back” asta va cauza o eroare.

3 Implementare

3.1 Diagrame de clase



3.2 Descriere detaliată



4 Bibliografie

<http://docwiki.embarcadero.com/>

<http://docwiki.embarcadero.com/RADStudio/Tokyo/en/VCL>

http://docwiki.embarcadero.com/RADStudio/Tokyo/en/HTML_Templates

<https://drive.google.com/file/d/1CsHQ80kRe6Mul4Tn7kcqM7ngbULnbnyZ/viewâ>

<http://bcu.ulbsibiu.ro/bibliografii/inginerie-calculatoare.htm>

Programare Orientata pe
Obiecte

C++ Builder WebBrower