**Browser Web**

**Airinei Vlad**

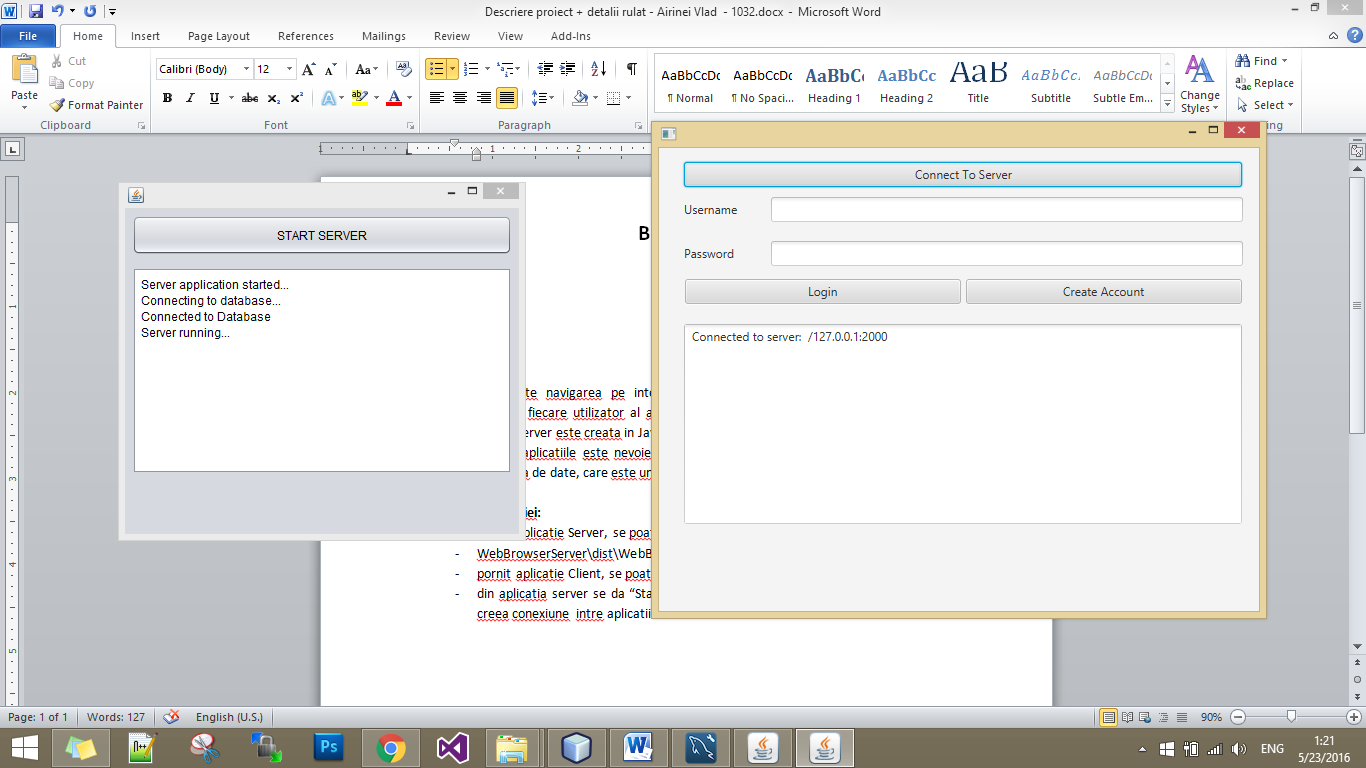
Grupa 1032

Aplicatia permite navigarea pe internet prin intermediul si pastreaza etichete de site-uri (bookmarks) pentru fiecare utilizator al acesteia. Aplicatia este formata din doua parti serverul si clientul, partea de server este creata in Java Swing iar cea de client cu ajutorul Java FX.

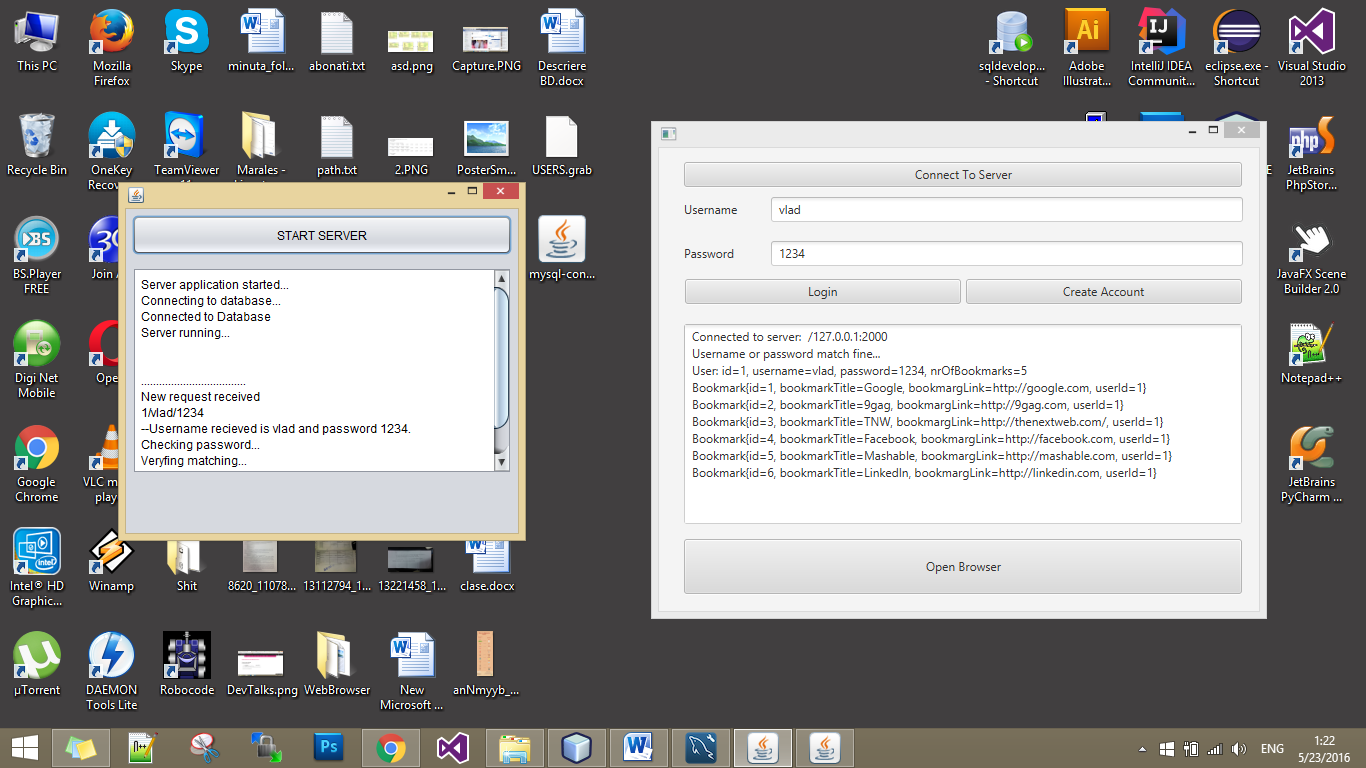
Pentru a rula aplicatiile este nevoie de acces la internet, atat pentru navigare cat si pentru connexiunea cu baza de date, care este una externa.

**Rularea Aplicatiei:**

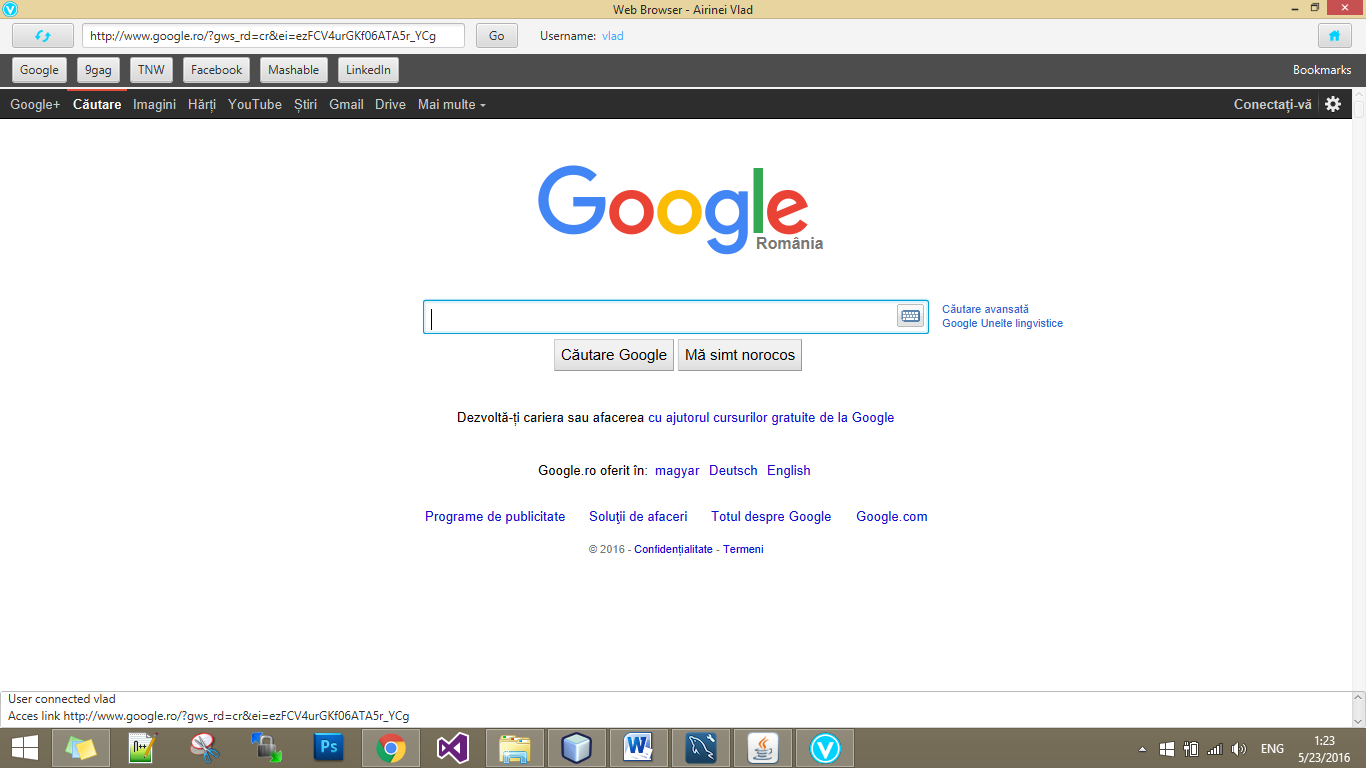
* pornit aplicatie Server, se poate face si din jar-ul gasit la :
* WebBrowserServer\dist\WebBrowserServer.jar
* pornit aplicatie Client, se poate face si din jar-ul gasit la : WebBrowser\dist\WebBrowser.jar
* din aplicatia server se da “Start Server”, iar din aplicatia client “Connect To Server” pentru a se creea conexiune intre aplicatii



* se introduce user: vlad si parola: 1234 si apoi se apasa buton Login



* in urma logarii va deveni vizibil butonul “Open Browser” ce se apasa pentru a rula browserul



**Aplicatia permite**

* crearea de nou conturi
* logarea cu conturi deja existente
* navigarea pe internet pe orice link introdus in campul pentru adrese
* refresh la pagina
* buton de home
* incarcarea din baza de date a etichetelor in functie de utilizator

**Despre aplicatie:**

Aplicatiile de server si client comunica printr-o conexiune de tip TCP/IP 1-1, cu ajutorul socket-ului.

Logarea se face prin verificarea in baza de date daca userul si parola coincid cu cele introduse in campurile din interfata grafica.

Daca logarea este realizata cu succes apare un buton care permite deschiderea browserului.

In browser etichetele (bookmarks-urile) sunt incarcate in functie de utilizator si se creeaza butoane in functie de cate bookmark-uri exista in baza de date. Pentru a testa asta sunt aici urmatoarele conturi ( user: admin pass: admin, user: root pass: root).

In partea de sus a aplicatiei exista buton de refresh care reincarca pagina actuala pe care se afla browserul.

TextFiled pentru introducerea adresei unde trebuie sa navigheze browserul. Aceasta are 2 handlere de eventuri, atunci cand se apasa de la tastatura butonul ENTER si atunci cand se modifica linkul introdus, daca acesta nu incepe cu http://”, acest prefix se adauga singur.

Butonul “Go” alternativa la ENTER, pentru a incarca pagina ce are linkul in TextField.

Username-ul care este conform cu cel introdus in ferestra de login, in concluzie datele se transmit intre cele doua ferestre, iar userul se pastreaza.

Butonul de “Home” care incarca automat “http://google.com” in TextField-ul pentru linkuri si invoca un event pentru butonul Go.

Bookmarks-urile sunt incarcate in functie de utilizator, numarul acestor butoane variind in functie de continutul bazei de date. Acestea sunt butoane ce incarca in TextField un anumit link si invoca event pentru butonul Go.

In partea centrala a paginii este prezent WebView-ul ce deserveste scopului principal al aplicatiei, navigarea pe internet.

Iar in partea de jos se gaseste un TextArea ce are rol de consola pentru a vedea ce actiuni sunt efectuate asupra WebView-ului.

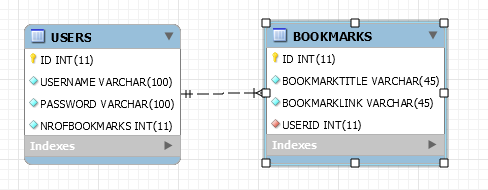
**Baza de date:** Este una mysql, hostata pe un server extern gratis pentru a putea rula aplicatia indiferent de device. Aceasta contine doua tabele USERS si BOOKMARKS cu urmatoarele atribute:

USERS :

* ID
* USERNAME
* PASSWORD
* NUMOFBOOKMARKS

BOOKMARKS

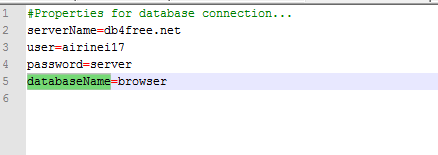
* ID
* BOOKMARKTITLE
* BOOKMARKLINK
* USERID



Connexiune la aceasta se face cu libraria **mysql connector**, importata in ambele proiecte.

**Fisier de configurare**

Datele de conectare pentru baza de date sunt obtinute prin incarcarea unui fisier de proprietati : **“bdProp.properties”** ce contine numele serverului, userul, parola si numele bazei de date



**Fisier log-uri**

Toate actiunile petrecute in cadrul aplicatiei de server se salveaza intr-un fisier de log-uri. Acestea contin diverse informatii despre fiecare interactiune cu aplicatia client, cat si un TIMESTAMP la sfarsit de rand pentru a stii cu exactitate cand a fost solicitata fiecare comanda.

**Puncte atinse din cerintele proiectului:**

1. Solutie distribuita, client-server 1-1
2. Thread-uri
3. Elementele de pe partea de server sa fie procesate multithread
4. Utilizare de fişiere de configurare (“**\WebBrowserServer\bdProp.properties**”)
5. Fişiere log (“**\WebBrowserServer\logs.txt**”)
6. JDBC + mysql
7. Java FX – grafica
8. Java Swing
9. Sockets
10. EventHandler
11. Clase de impachetare