UNIVERSITATEA TEHNICĂ ”GHEORGHE ASACHI” IAȘI

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

**PLATFORMĂ ONLINE DE CATERING**

***TI-P***

**Echipa:**

*Bărănceanu Șerban (liderul echipei)*

*Chirilă Simona Mariana*

*Popa Ioan-Dragoș*

Grupa 1307A

2018

**Tema și motivarea alegerii**

Tema proiectului constă în crearea unei platforme online ce oferă servicii de catering. Aplicația dezvoltată de noi permite posibilitatea de a crea un cont de utilizator prin care aceștia au acces la “bucătăria” platformei. Aceasta este pusă la dispozoție de colaboratorii noștrii, carora le oferim, de asemenea, posibilitatea de a-și prezenta preparatele gastronomice făcute în propria casă sau în laborator. Utilizatorii pot comanda, oferi review-uri și interacționa direct cu producătorii înregistrați pe site-ul nostru.

Am ales această temă deoarece considerăm că trebuie sa le oferim o șansă pasionaților de bucătărie să interacționeze unii cu ceilalți, să-și îmbogățească cunoștințele și să-și promoveze produsele. De asemenea, am ales această temă pentru că ne place mâncarea bună ☺.

***Alte*** ***aplicații existente*** în mediul online sunt site-urile marilor magazine, cum ar fi Mamma Mia, Dopo Poco, Class, KFC ș.a. ce oferă posibilitatea de a comanda, însă doar produse proprii; proiectul nostru aduce o noutate în ceea ce privește serviciul de catering.

**Descrierea generală a implementării**

Aplicația dezvoltată este una web și are ca scop crearea unui mediu în care utilizatorul se poate loga în contul creat, poate alege din produsele expuse sau poate să-și expună propriile produse.

Un utilizator își va putea crea un cont prin completarea unui formular, după care se va putea autentifica în contul său prin intermediul unui nume de utilizator și a unei parole. Odata logat în contul sau, utilizatorul poate să vizualizeze toate ofertele de produse oferite de site-ul nostru, poate vedea informații suplimentare despre fiecare articol în parte și se poate hotărâ daca dorește să plaseze o comandă. Acesta poate adăuga toate produsele dorite în coșul său de cumparături și prin completarea unui formular, poate trimite comanda. De asemenea, fiecare utilizator are posibilitatea de a-și posta produsul său propriu, devenind în acest mod producator, putănd astfel onora comenzi de la alti utilizatori ai site-ului. Utilizatorii pot oferi review-uri la fiecare produs, ajutând astfel la imbunatățirea serviciilor.

Site-ul poate fi accesat și fără un cont de utilizator. În acest mod doar se pot vizualiza anunțurile cu produse și informații generale despre site, fără a avea posibilitatea de a plasa o comandă sau a oferi un review pentru un produs.

**Resursele hardware/software utilizate**

Aplicația va fi disponibilă pe orice navigator web modern, pe desktop și pe mobile. Resurele hardware folosite sunt 3 laptop-uri cu procesoare intel și sistem de operare Windows 10. Resursele software folosite sunt:

- editorul de text Notepad++

- IDE-ul Eclipse

- web browser: Chrome, Firefox și Edge

- web server: Apache Tomcat

- sistem de versionare: Github

- server bază de date: MySQL

**Actori și componente principale**

În aplicatia web dezvoltată regăsim 2 actori, *vizitator* şi *utilizator*. Actorul *vizitator* devine *utilizator* în urma înregistrării, iar *utilizatorul* neautentificat este considerat *vizitator*. Diferenţa între actori o reprezintă privilegiile fiecăruia in cadrul site-ului, *utilizatorul* putând accesa mai multe componente principale.

Componentele principale ale site-ului sunt prezentate de urmatoarele: *înregistrare*, *autentificare*, *despre*, *vizualizare meniu*, *adaugă produs*, *plasare comandă*.

Secțiunea *înregistrare* oferă posibilitatea *vizitatorului* de a-şi creea un cont şi de a deveni *utilizator*. Prin componenta *autentificare,* *utilizatorul* are posibilitatea de a-și accesa contul.

Secțiunea *despre* oferă o scurtă prezentare asupra site-ului, iar *vizualizare meniu* oferă posibilitatea de a afla produsele disponibile la acel moment.

Componenta *adaugă produs* oferă posibilitatea *utilizatorilor* de a-și înregistra produsul pe site-ul nostru.

Componenta *plasare comandă* reprezintă metoda prin care un utilizator își poate procura produsele dorite prin intermediul unui formular de comandă.

**Diagrama UML Use-Case**

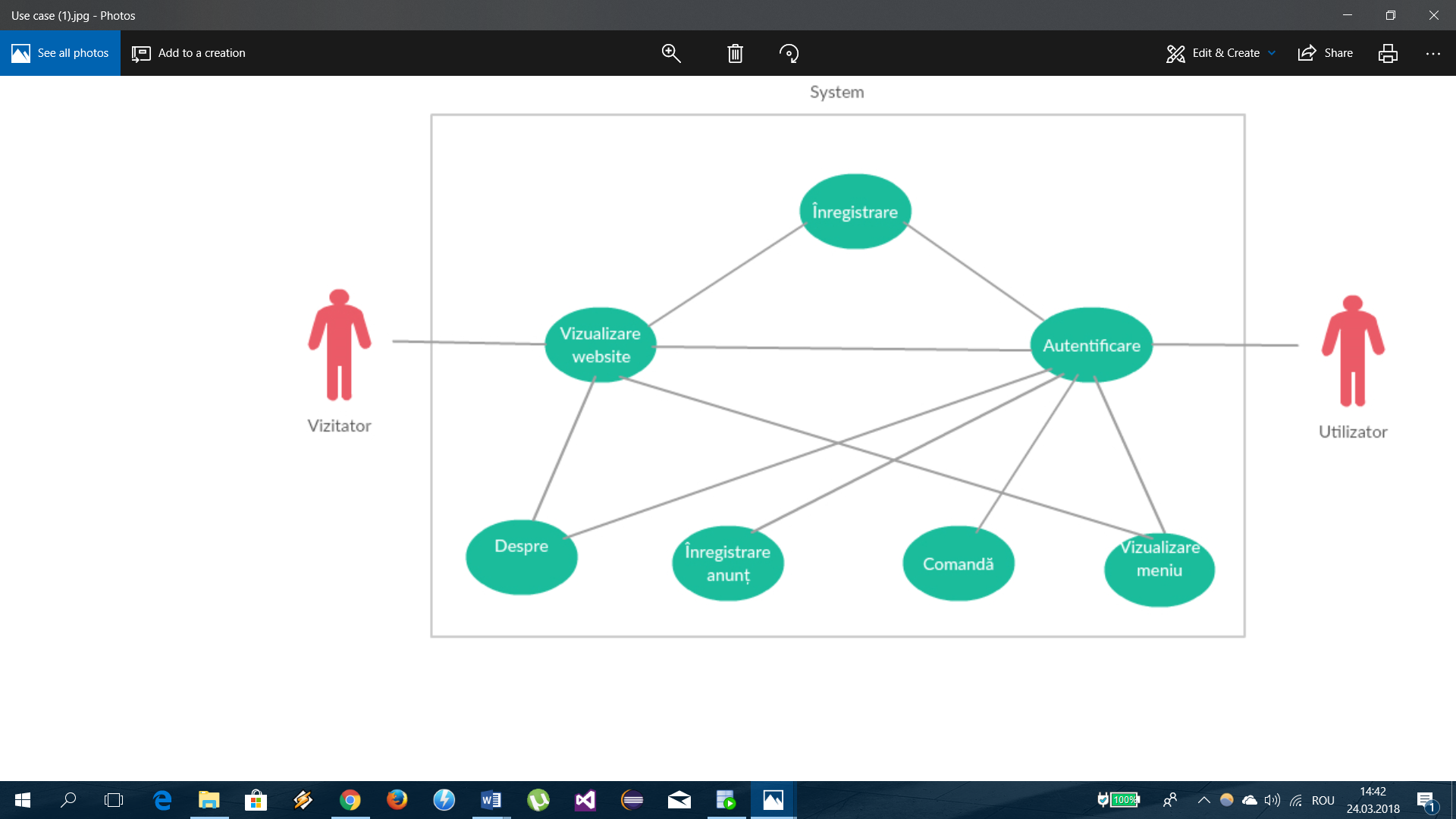
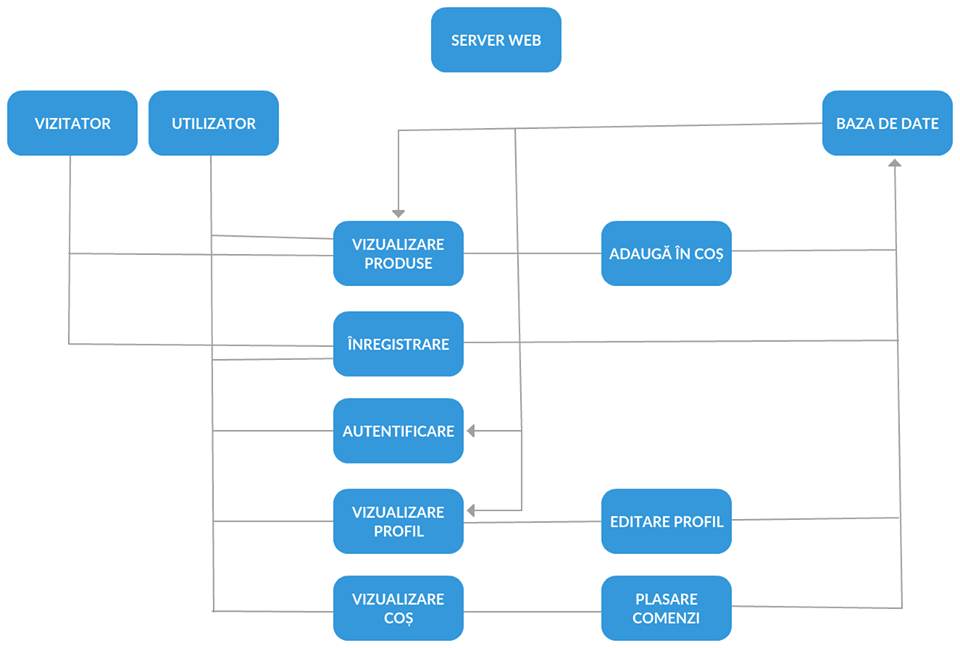


Diagrama UML de use-case prezintă cele două tipuri de utilizatori ce pot accesa site-ul nostru și acțiunile pe care fiecare dintre ei le pot intreprinde pe site.

Vizitatorul este utilizatorul fără un cont activ, iar acesta are posibilitatea să vizualizeze intreg site-ul, poate, dacă dorește, să își facă un cont și poate să citească mai multe informații despre site-ul nostru și despre menirea acestuia la secțiunea Despre.

Un utilizator cu un cont activ are posibilitatea să se autentifice. După autentificare are posibilitatea să adauge în coș produsele dorite de la diferiți producatori, și în urma completării datelor, să plaseze o comandă. Acesta mai are posibilitatea să își plaseze propriul anunț sau poate să analizeze alte anunțuri și să ofere review-uri, ce sperăm că vor duce la creșterea calității serviciilor oferite.

**Diagrama UML information flow**

****

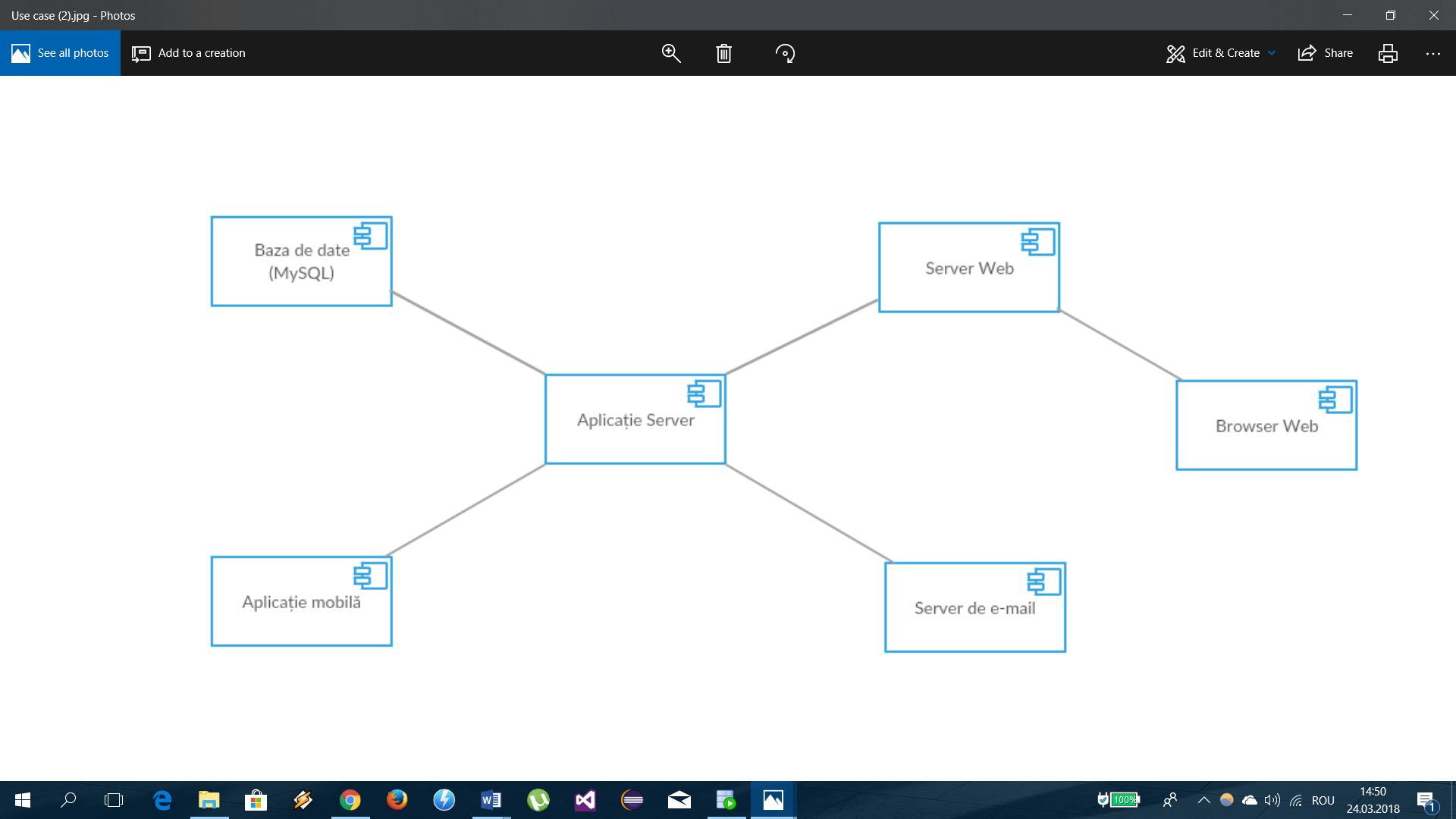
Atunci când cei doi actori accesează *vizualizare produse* se extrag din baza de date informaţii care sunt prelucrate si afişate.

La accesarea componentei *Înregistrare* de către *vizitator*, în urma unor validări, sunt adăugate informaţiile legate de cont în baza de date şi se transmite un email. La cererea de *autentificare* a *utilizatorului* se accesează baza de date şi sunt verificate informaţiile de autentificare, iar în urma procesării se realizează autentificarea sau nu.

Atunci când *utilizatorul* accesează *vizualizare profil* se extrag din baza de date informaţiile legate de contul *utilizatorului*, cum ar fi adresa, imaginea de profil, etc. şi sunt afişate. Din cadrul acestei componente, *utilizatorul* are posibilitatea de a-şi edita profilul, moment în care se adaugă în baza de date noile informații.

În cadrul secțiunii *vizualizare coș* se poate merge mai departe către *plasarea comenzii*.

**Diagrama UML de componente**



Această diagram prezintă principalele componente utilizate de aplicația noastră Toate aceste componente sunt conectate la server și, astfel, intre ele. În baza de date se stochează informații cu privire la conturile utilizatorilor, informații despre anunțurile existente pe site și datele comenzilor plasate.

Browser-ul Web pune la dispoziția utilizatorului tot conținutul site-ului. Serverul mail oferă suport pentru serviciile de mail, iar aplicația mobilă oferă utilizatorilor posibilitatea de a vizualiza site-ul si de pe un dispozitiv mobil.

**Diagrama UML de activitate**

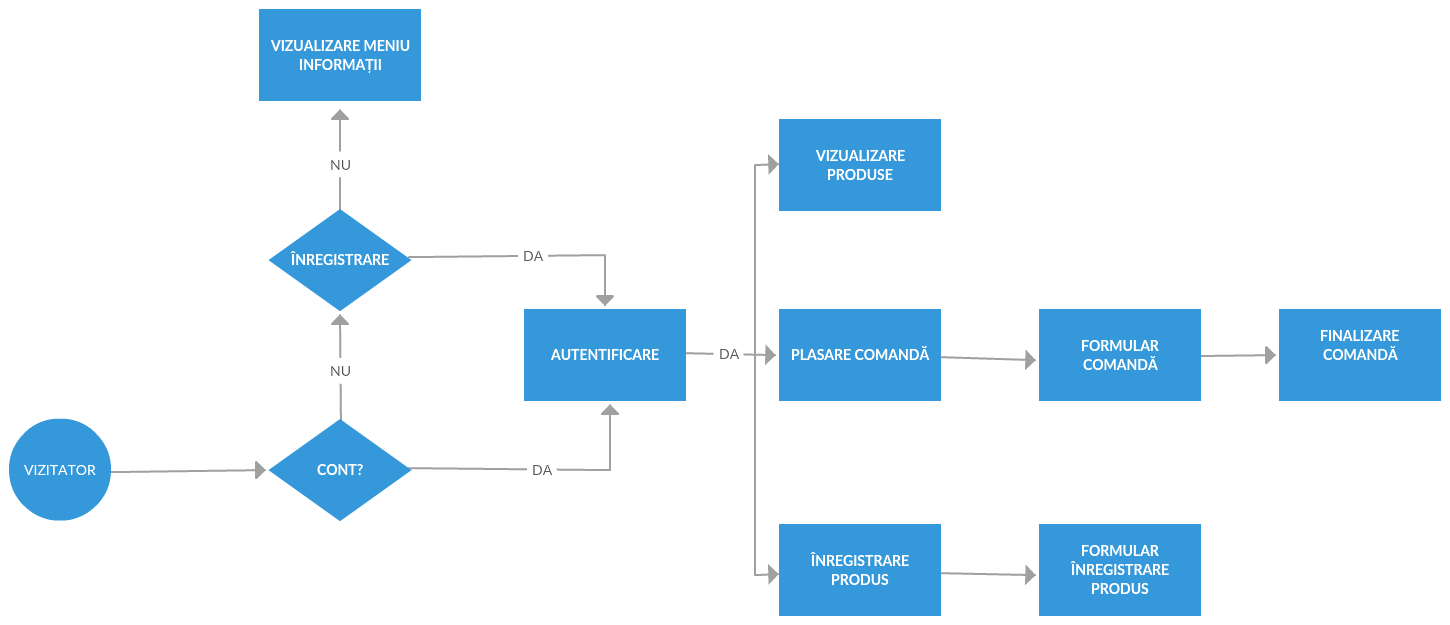
****

Diagrama UML de activitate infățișează acțiunile și activitățile ce pot fi executate de un utilizator pe site și modul în care poate fi utilizată aplicația.

La intrarea pe site, utilizatorul are posibilitatea să își facă un cont sau să se logheze în contul său existent deja. Daca alege să continuie fara un cont, utilizatorul poate să vizualizeze anunțurile existente pe site, poate citi informații despre site și activitatea acestuia, neavând,însa, posibilitatea de a comanda mâncare sau să ofere review-uri.

Dacă utilizatorul se autentifică în contul său, poate, în acest moment, să adauge preparate în coșul său, poate plasa o comandă de mâncare, poate înregistra propriul său anunț și are posibilitatea să lase review-uri altor utilizatori sau altor anunțuri.

**Diagrama UML de secvență**

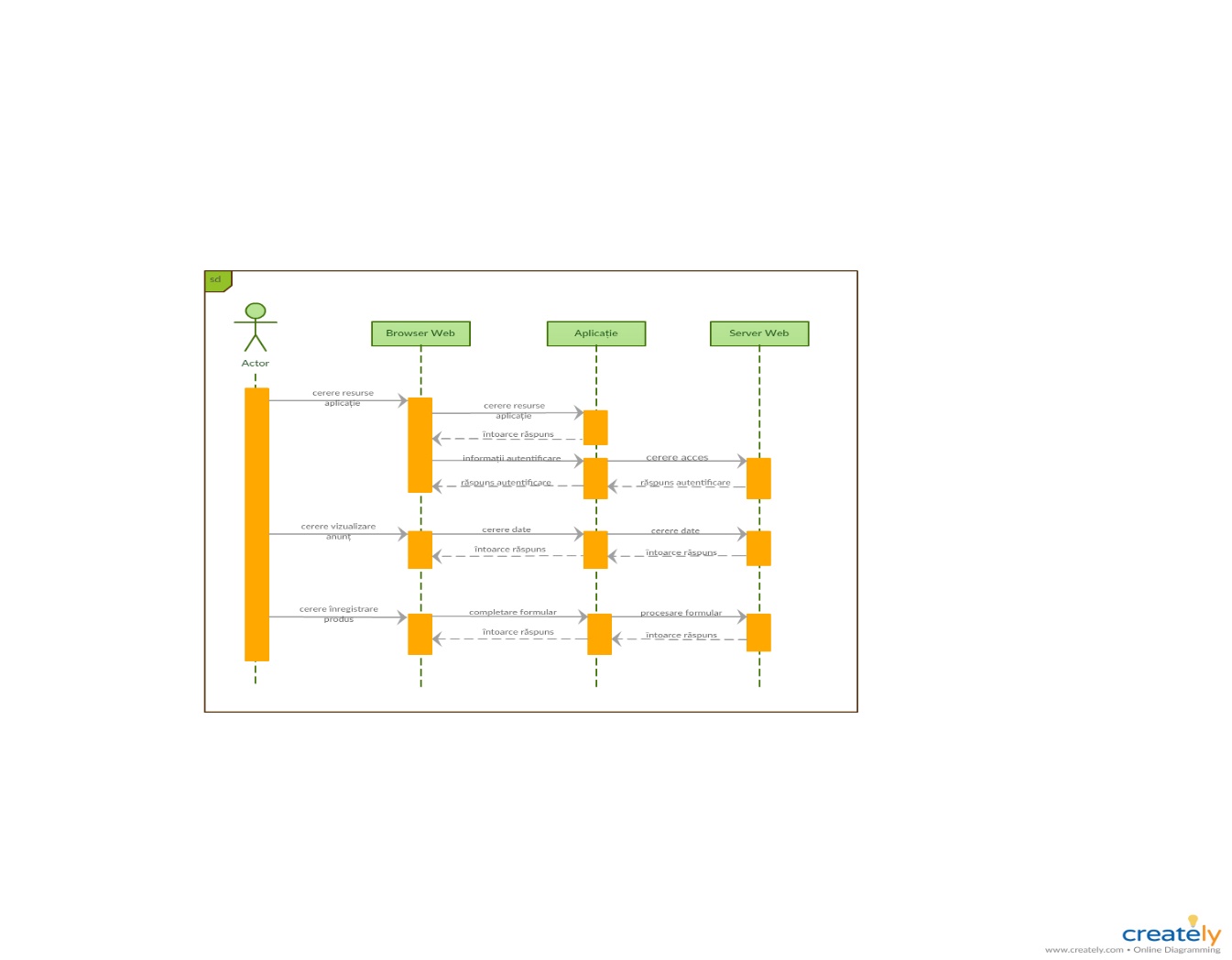
****

Diagrama UML de secvență este folosită pentru a descrie interacțiunea dintre un utilizator, aplicația propriu-zisă și serverul utilizat de către aplicație .

Utilizatorul interacționează prin intermediul unui browser web cu aplicația noastră, trimițând o cerere aplicației pentru a primi date. În primă instanță, utilizatorul are acces doar la anumite date ale aplicației, restul datelor fiind accesibile în momentul în care utilizatorul se autentifică pe site. Astfel, browser-ul web trimite o cerere pentru a accesa restul datelor prin sistemul de autentificare, numele de utilizator și parola fiind validate de către server.

Dacă datele de autentificare transmise de către browser sunt validate, atunci serverul permite accesul utilizatorului și transmite restul datelor, având dreptul de a adăuga produs/plasa comandă, iar dacă datele nu corespund atunci va apărea un mesaj de eroare de la server.

**Diagrama UML de stare**

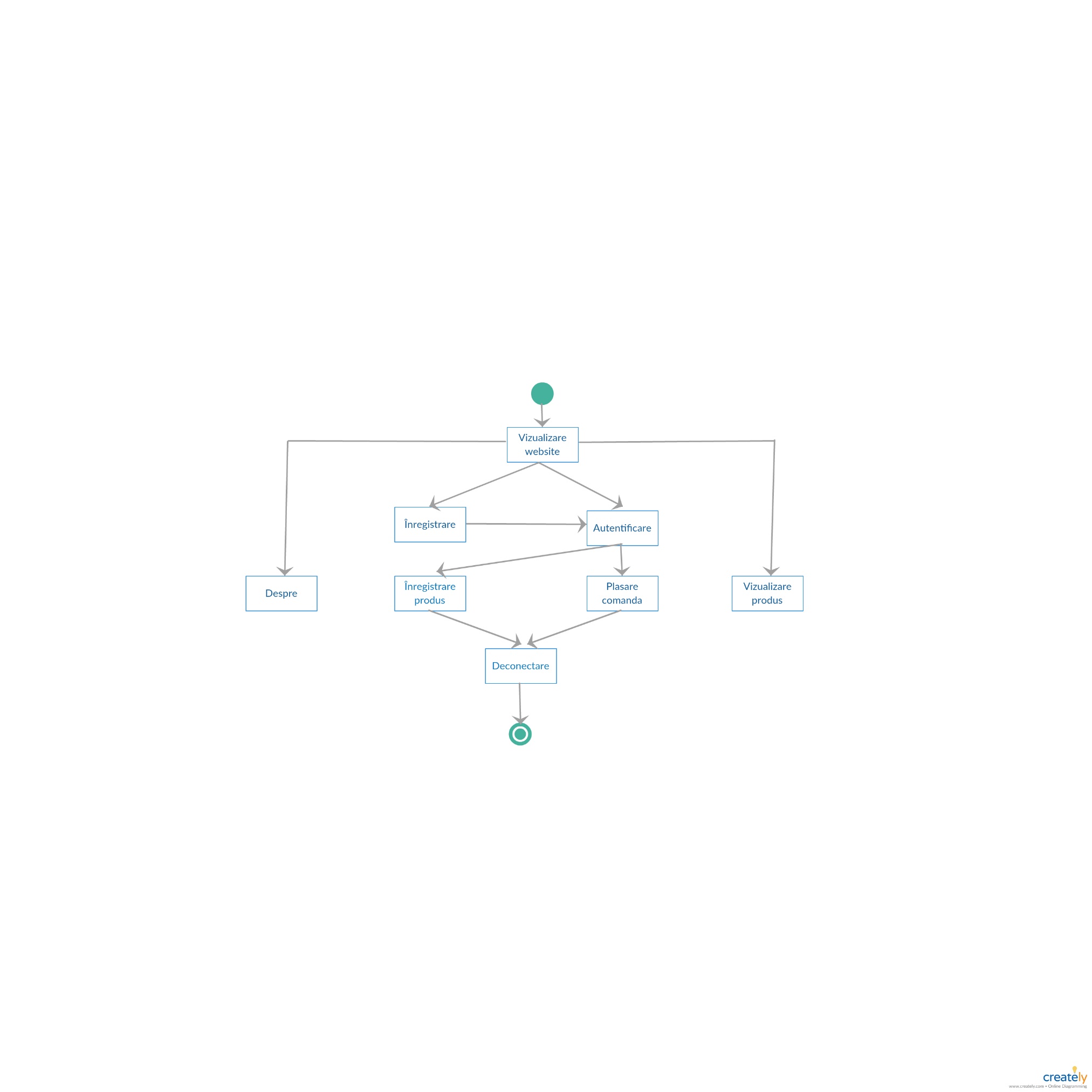
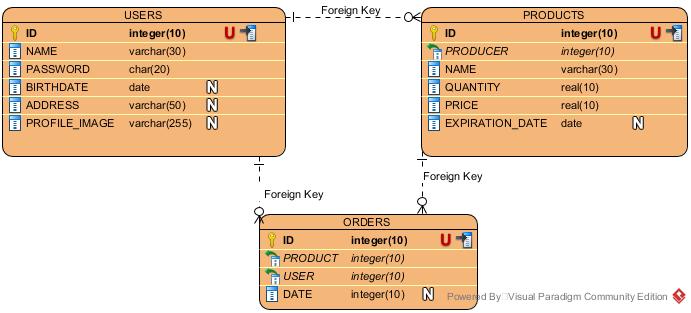
****

Diagrama de stare modelează viața aplicației implementate din momentul în care un utilizator accesează site-ul până în momentul în care acesta părăsește site-ul.

Astfel, punctul inițial este momentul în care un utilizator accesează site-ul . Acesta poate trece în stările de vizionare a paginilor cu informații sau în starea de autentificare, respectiv înregistrare a utilizatorilor. În momentul în care se află în starea de autentificare, după aceasta poate trece în stările de vizionare a paginilor informative sau în starea de înregistrare produs sau plasare comandă. După ce a înregistrat un produs sau în momentul terminării plasării comenzii, utilizatorul poate alege să se deconecteze, ceea ce reprezintă starea finală a aplicației.

**Diagrama Entity-Relationship pentru baze de date**

****

Baza de date conține 3 tabele: USERS, PRODUCTS și ORDERS. În tabela USERS vor fi stocate datele tuturor utilizatorilor înregistrați. În tabela PRODUCTS vor fi stocate datele produselor publicate pe site. Ea va fi legată de utilizatori prin coloana PRODUCER, În tabela ORDERS vor fi stocate comenzile efectuate pe site. Această tabelă este legată de USERS prin coloana USER și de PRODUCTS prin coloana PRODUCT. Toate tablele au ca și index o cheie primară ID care este unică.

**Ce s-a realizat până acum**

Aplicația web a fost dezvoltată ca un SPA (Single Page Application) care încarcă o singură pagină JSP (index.jsp) ce prezintă un formular de logare în contul de utilizator. Odată logat utilizatorul poate vizualiza într-un tabel produsele disponibile, produse ce sunt preluate din baza de date din tabela products.

Am reușit, până în acest punct, să realizăm conexiunea la o bază de date în care am creat 2 tabele, users si products, baza de date fiind interogată de un server de servicii web. Am utilizat 2 metode HTTP diferite, GET și POST.

Am utilizat un server web care folosește un servlet si un server web ce folosește un JSP. Pentru realizarea design-ului am utilizat HTML5, CSS și JavaScript.

**Ce urmează a fi implementat**

Ne propunem ca în etapa următoare să adăugam funcționalități site-ului nostru. Un utilizator își va putea crea un cont prin completarea unui formular, după care se va putea autentifica în contul său prin intermediul unui nume de utilizator și a unei parole.

Site-ul va putea fi accesat și fără un cont de utilizator. În acest mod doar se vor vizualiza anunțurile cu produse și informații generale despre site, fără a avea posibilitatea de a plasa o comandă sau a oferi un review pentru un produs.

Odată logat într-un cont, se vor putea vizualiza toate ofertele de produse oferite de site-ul nostru. Totodată, acesta va putea vedea informații suplimentare despre fiecare articol în parte și se va poate hotărâ daca dorește să plaseze o comandă. Acesta poate adăuga toate produsele dorite în coșul său de cumparături și prin completarea unui formular, poate trimite comanda. De asemenea, fiecare utilizator are posibilitatea de a-și posta produsul său propriu într-un anunț, devenind în acest mod producator, putănd astfel onora comenzi de la alti utilizatori ai site-ului. Utilizatorii pot oferi review-uri la fiecare produs, ajutând astfel la imbunatățirea serviciilor.

**Dificultăți întâmpinate**

Câteva dintre problemele întâmpinate în realizarea acestui site web au fost conectarea servlet-urilor cu server-ul de servicii web și utilizarea metodelor oferite de acesta din urmă și parsarea răspunsului în format json returnat de server-ul de servicii web în servlet și afișarea produselor din baza de date în pagina web.

**Contribuția fiecărui membru al echipei**

Șerban, liderul echipei, s-a ocupat de crearea baza de date și a tabelelor, de server-ul de servicii web și de managementul proiectului in GitHub. Totodata, el a oferit support celorlalți membri ai echipei când aceștia întampinau o dificultate.

Dragoș s-a ocupat de crearea servleturilor, și de crearea paginii index.jsp și de realizarea documentațiilor.

Simona a realizat aplicația desktop, a lucrat la paginile Home.jsp și Produse.jsp, contribuind și la realizarea documentației.

**Evaluarea membrilor echipei**

**Simona**

Simona a fost foarte deschisă la tot ce s-a propus în cadrul echipei și a fost mereu doritoare să participe activ în fiecare aspect al implementării proiectului. A fost foarte energică și comunicativă, venind mereu cu idei noi și cu rezolvări la probelemele ce mai apăreau pe parcursul implementării.

**Dragoș**

Dragoș a lucrat foarte bine în echipă, a venit mereu cu idei noi și cu soluții pentru rezolvarea problemelor. A fost foarte activ, a fost mereu alături de colegii de echipă și s-a ocupat foarte bine de sarcinile ce îi erau date. Dacă se bloca și nu stia să rezolve o anumită problem, cerea ajutorul nostru, și cu ajutorul echipei reuseam să rezolvăm orice problemă ivită.

**Evaluarea liderului de c**ă**tre echip**ă

**Simona**

Șerban a fost cel mai bun lider. Șerban s-a ocupat foarte bine de coordonarea echipei, ne-a trasat fiecăruia dintre noi sarcini clare și de fiecare dată ne ajuta când ne impotmoleam sau nu știam cum să rezolvăm o anumită problemă apărută pe parcursul dezvoltării proiectului. A fost foarte deschis la sugestii și a luat mereu în calcul ce îi sugeram și noi.

**Dragoș**

Șerban și-a asumat foarte bine atribuțiile de lider și s-a comportat mereu în consecință. A fost mereu alături de noi cu sfaturi și cu îndrumări. Ne-a împărțit în mod echilibrat task-urile și a organizat periodic întalniri în care am discutat și am implementat, ca o adevărata echipă, ce lucruri ne propuneam. Șerban a venit cu foarte multe surse de informații care ne-au ajutat la întelegerea și implementarea propriu-zisă.

**Concluzii și modalități de extindere a aplicației**

Proiectul a fost foarte solicitant pentru că nu am mai lucrat cu servelet-uri, jsp-uri, github. Am învățat cum să lucrăm cu aceste tehnologii.

Aplicația ar putea fi extinsă prin completarea cu noi funcționalități, cum ar fi adăugarea unui coș de cumpărături și a unui sistem de logare împreună cu un profil personalizat pentru fiecare utilizator.

**Bibliografie**

<https://www.geeksforgeeks.org/parse-json-java/>

<https://www.tutorialspoint.com/servlets/index.htm>

<https://www.tutorialspoint.com/jsp/index.htm>

<https://examples.javacodegeeks.com/desktop-java/swing/java-swing-application-example/>

<http://restsharp.org/>