

PROIECT – INGINERIA SISTEMELOR SOFTWARE

AN UNIVERSITAR 2022-2023

SEMESTRUL 2

Aplicatie

Agentie de vanzari

Nume si Prenume Student

Pâslaru Gabriel

Grupa 225/2

Sectia Informatica romana

Coordonator

Prof. Sima Ioan

1. Cerintele aplicatiei

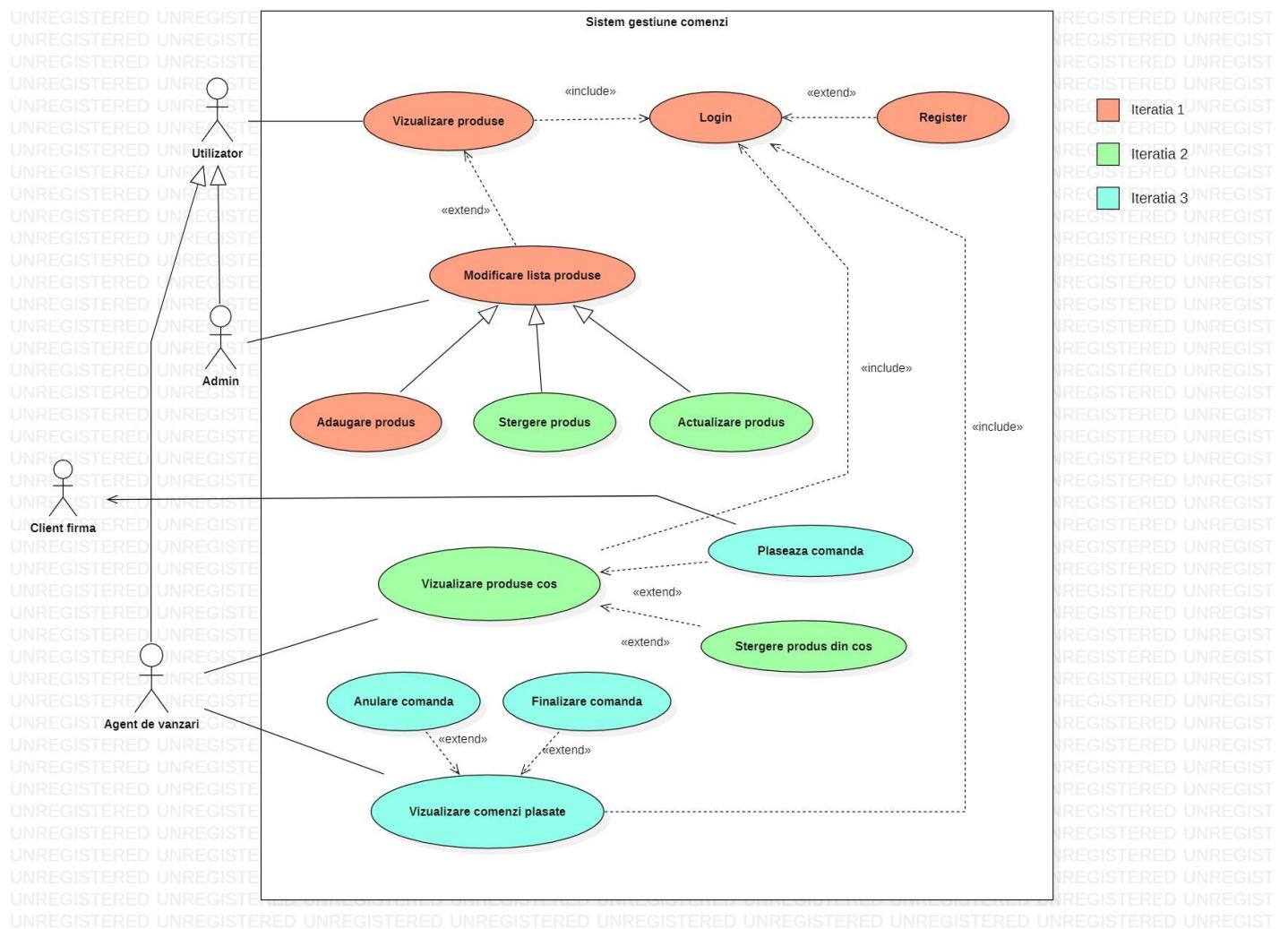
AGENTI DE VANZARI

O firma are mai multi agenti de vanzari, care fac comenzi pentru clientii firmei si le distribuie acestora produsele. Pentru desfasurarea activitatilor, firma ofera o aplicatie prin care agentii pot comanda produse. Aplicatia pune la dispozitia fiecarui agent al firmei un terminal prin care:

- agentul vizualizeaza lista tuturor produselor vândute de firma, împreuna cu preturile aferente și cantitatile existente pe stoc.
- agentul poate comanda o cantitate dintr-un produs. După orice comanda valida, toți agentii logați în aplicație vor vedea lista actualizată a stocurilor (este posibil ca declansarea înregistrării unei comenzi din partea unui agent să conduca la un mesaj informativ "cantitate insuficientă în stoc").

2. Modelul funcțional

2.1 Diagrama cazurilor de utilizare



2.2 Descrierea cazurilor de utilizare

CU-1: Login

ID and name	CU-1: Login		
Primary actor	Utilizatorul	Secondary actors	-
Description	Utilizatorul se autentifica in aplicatie.		
Trigger	Utilizatorul da dublu click pe aplicatia de gestiune a comenzilor.		
Preconditions	PRE-1. Utilizatorul are un cont existent.		
Postconditions	POST-1. Utilizatorul se autentifica in aplicatie. POST-2. Se afiseaza principala pagina a aplicatiei de unde utilizatorul poate efectua diverse operatii.		
Normal flow	1.0 Login <ol style="list-style-type: none"> 1. Utilizatorul da dublu click pe iconita aplicatiei. 2. Sistemul afiseaza fereastra pentru autentificare. 3. Utilizatorul isi introduce username-ul. 4. Utilizatorul isi introduce parola. 5. Utilizatorul trimit datele de autentificare sistemului prin apasarea butonului de login. 6. Sistemul afiseaza fereastra cu meniul principal al aplicatiei. 		
Alternative flows	1.1 Utilizatorul nu are cont <ol style="list-style-type: none"> 1. Utilizatorul acceseaza meniul de register. 2. Se declanseaza cazul CU-2. 		
Exceptions	1.0.E1 Datele introduse de utilizator sunt gresite <ol style="list-style-type: none"> 1. Sistemul avertizeaza utilizatorul ca datele introduse nu corespund unui cont existent. 2. Se revine la normal flow pasul 1.0.2. 		

CU-2: Register

ID and name	CU-2: Register		
Primary actor	Utilizatorul	Secondary actors	-
Description	Utilizatorul creeaza un cont nou in cadrul aplicatiei.		
Trigger	Utilizatorul da click pe butonul de register din fereastra de autentificare.		
Preconditions	Aplicatia este deschisa si afiseaza fereastra de autentificare.		
Postconditions	POST-1. Se creeaza un nou cont pentru utilizator in cadrul sistemului. POST-2. Utilizatorul va fi autentificat in aplicatie.		
Normal flow	<p>2.0 Register</p> <ol style="list-style-type: none"> 1. Utilizatorul da click pe butonul de register din fereastra de autentificare. 2. Sistemul afiseaza fereastra de register. 3. Utilizatorul isi introduce datele pentru crearea unui nou cont. 4. Utilizatorul trimit datele sistemului prin apasarea butonului de register. 5. Sistemul creeaza un nou cont pentru utilizator ce corespunde datelor introduse. 6. Sistemul afiseaza fereastra cu meniul principal al aplicatiei. 		
Alternative flows	<p>2.1 Utilizatorul are deja un cont existent</p> <ol style="list-style-type: none"> 1. Utilizatorul acceseaza meniul de login. 2. Se declanseaza cazul CU-1. 		
Exceptions	<p>2.0.E1 Username-ul introdus corespunde unui cont existent</p> <ol style="list-style-type: none"> 1. Sistemul informeaza utilizatorul ca username-ul este deja folosit. 2. Se revine la normal flow pasul 2.0.2. <p>2.0.E1 Numele introduse nu este valid</p> <ol style="list-style-type: none"> 1. Sistemul avertizeaza utilizatorul ca datele introduse nu sunt valide. 2. Se revine la normal flow pasul 2.0.2. 		

CU-3: Vizualizare produse

ID and name	CU-3: Vizualizare produse		
Primary actor	Utilizatorul	Secondary actors	-
Description	Utilizatorului i se prezinta lista cu toate produsele firmei impreuna cu pretul fiecaruia si cantitatea produsului aflata in stoc.		
Trigger	Autentificarea cu succes a utilizatorului in aplicatie.		
Preconditions	PRE-1. Utilizatorul este logat.		
Postconditions	POST-1. Lista cu toate produsele este afisata.		
Normal flow	<p>3.0 Vizualizare produse</p> <p>1. Dupa CU-1 sistemul afiseaza in fereastra principala a aplicatiei lista cu produsele impreuna cu pretul lor si cantitatea disponibila in stoc.</p>		
Alternative flows	-		
Exceptions	-		

CU-4: Adaugare produs

ID and name	CU-4: Adaugare produs		
Primary actor	Administratorul	Secondary actors	-
Description	Administratorul adauga un nou produs in lista de produse oferite spre vanzare de catre firma, adaugand un nume pentru produs, o scurta descriere, un pret si o cantitate initiala de stoc.		
Trigger	Administratorul da click pe butonul Add product din fereastra de gestiune a produselor.		
Preconditions	<p>PRE-1. Administratorul este autentificat.</p> <p>PRE-2. Contul este unul valid cu functia de administrator.</p> <p>PRE-3. Fereastra pentru gestiunea produselor este deschisa.</p>		

Postconditions	<p>POST-1. Se adauga un nou produs cu numele, descrierea, pretul si cantitatea specificata.</p> <p>POST-2. Se actualizeaza lista cu produsele oferite de firma.</p>
Normal flow	<p>4.0 Adaugare produs</p> <ol style="list-style-type: none"> 1. Administratorul de click pe butonul de adaugare produs. 2. Sistemul afiseaza meniul pentru adaugarea unui produs. 3. Administratorul introduce numele produsului. 4. Administratorul introduce descrierea produsului. 5. Administratorul introduce pretul produsului. 6. Administratorul introduce cantitatea initiala din stoc a produsului. 7. Administratorul trimit datele despre produs sistemului prin apasarea butonului de add. 8. Sistemul informeaza administratorul ca produsul s-a adaugat cu succes. 9. Sistemul inchide meniul pentru adaugarea unui produs. 10. Sistemul actualizeaza lista de produse afisata.
Alternative flows	-
Exceptions	<p>4.0.E1 Numele produsului nu este valid</p> <ol style="list-style-type: none"> 1. Sistemul avertizeaza utilizatorul ca datele introduse nu sunt valide. 2. Se revine la normal flow pasul 4.0.2. <p>4.0.E2 Pretul introdus nu este un numar mai mare decat zero</p> <ol style="list-style-type: none"> 1. Sistemul avertizeaza utilizatorul ca datele introduse nu sunt valide. 2. Se revine la normal flow pasul 4.0.2. <p>4.0.E3 Cantitatea nu este un numar mai mare decat zero</p> <ol style="list-style-type: none"> 1. Sistemul avertizeaza utilizatorul ca datele introduse nu sunt valide. 2. Se revine la normal flow pasul 4.0.2.

CU-5: Stergere produs

ID and name	CU-5: Stergere produs		
Primary actor	Administratorul	Secondary actors	-
Description	Administratorul selecteaza un produs din lista de produse oferite spre vanzare de catre firma si il sterge definitiv din stoc.		
Trigger	Administratorul selecteaza un produs din lista de produse si apasa butonul de stergere.		
Preconditions	PRE-1. Administratorul este autentificat. PRE-2. Contul este unul valid cu functia de administrator. PRE-3. Fereastra pentru gestiunea produselor este deschisa.		
Postconditions	POST-1. Se sterge un produs din lista. POST-2. Se actualizeaza lista cu produsele oferite de firma.		
Normal flow	5.0 Stergere produs <ol style="list-style-type: none"> Administratorul selecteaza produsul din lista. Administratorul trimitere cererea de stergere sistemului prin apasarea butonului de stergere produs. Sistemul informeaza administratorul ca produsul s-a sters cu succes. Sistemul actualizeaza lista de produse afisata. 		
Alternative flows	5.1 Stergere multiple produse <ol style="list-style-type: none"> Administratorul selecteaza produsele din lista. Se revine la normal flow pasul 5.0.2. 		
Exceptions	5.0.E1 Stergerea s-a actionat fara selectarea unui produs <ol style="list-style-type: none"> Sistemul avertizeaza utilizatorul ca trebuie sa selecteze un produs prima data. Se revine la normal flow pasul 5.0.1. 		

CU-6: Actualizare produs

ID and name	CU-6: Actualizare produs		
Primary actor	Administratorul	Secondary actors	-
Description	Administratorul selecteaza un produs din lista de produse oferite spre vanzare de catre firma pe care doreste sa il actualizeze. Se deschide un meniu in care administratorul adauga noul nume pentru produs, o noua descriere si un nou pret.		
Trigger	Administratorul indica faptul ca doreste actualizarea unui produs existent in stoc.		
Preconditions	PRE-1. Administratorul este autentificat. PRE-2. Contul este unul valid cu functia de administrator. PRE-3. Fereastra pentru gestiunea produselor este deschisa.		
Postconditions	POST-1. Pentru un produs din lista se actualizeaza numele, descrierea si pretul. POST-2. Se actualizeaza lista cu produsele oferite de firma.		
Normal flow	6.0 Actualizare produs <ol style="list-style-type: none"> 1. Administratorul selecteaza produsul din lista de produse. 2. Sistemul afiseaza fereastra pentru actualizarea unui produs. 3. Administratorul introduce numele produsului. 4. Administratorul introduce descrierea produsului. 5. Administratorul introduce pretul produsului. 6. Administratorul introduce stocul produsului. 7. Administratorul trimit datele despre produs sistemului prin apasarea butonului de actualizare produs. 8. Sistemul informeaza administratorul ca produsul s-a actualizat cu succes. 9. Sistemul inchide meniul pentru actualizarea unui produs. 10. Sistemul actualizeaza lista de produse afisata. 		
Alternative flows	-		

Exceptions	6.0.E2 Pretul introdus nu este un numar mai mare decat zero <ul style="list-style-type: none"> 1. Sistemul avertizeaza utilizatorul ca datele introduse nu sunt valide. 2. Se revine la normal flow pasul 6.0.5.
-------------------	---

CU-7: Vizualizare produse cos

ID and name	CU-7: Vizualizare produse cos		
Primary actor	Agentul de vanzari	Secondary actors	-
Description	Agentului de vanzari i se prezinta lista cu produsele firmei impreuna cu cantitatea si pretul individual al fiecarui produs aflat in cos.		
Trigger	Agentul de vanzari declanseaza afisarea produselor din cos.		
Preconditions	PRE-1. Agentul de vanzari este autentificat. PRE-2. Contul este unul valid cu functia de agent de vanzari. PRE-3. Fereastra principala a agentului de vanzari este afisata.		
Postconditions	POST-1. Se afiseaza lista cu toate produsele din cos ce corespund comenzi ce urmeaza a fi lansata.		
Normal flow	7.0 Vizualizare produse cos <ul style="list-style-type: none"> 1. Sistemul afiseaza cosul de cumpărături cu fiecare produs, cantitate selectată și preț individual. 		
Alternative flows	-		
Exceptions	-		

CU-8: Stergere produs din cos

ID and name	CU-8: Stergere produs din cos		
Primary actor	Agentul de vanzari	Secondary actors	-
Description	Agentul de vanzari selecteaza un produs din lista de produse aflate in cos si il sterge definitiv din acesta (il elibera din comanda ce urmeaza a fi lansata).		
Trigger	Agentul de vanzari selecteaza produsul si apasa pe butonul de stergere.		
Preconditions	PRE-1. Agentul de vanzari este autentificat. PRE-2. Contul este unul valid cu functia de agent de vanzari. PRE-3. Fereastra principala a agentului de vanzari este afisata. PRE-4. Cosul contine cel putin un produs.		
Postconditions	POST-1. Se sterge produsul din lista de produse selectate. POST-2. Se actualizeaza lista cu produsele din cos.		
Normal flow	8.0 Stergere produs din cos <ol style="list-style-type: none"> Agentul de vanzari selecteaza produsul pe care vrea sa il stearga din cos. Agentul de vanzari declanseaza stergerea prin apasarea butonului de stergere produs. Sistemul informeaza agentul de vanzari ca produsul s-a sters din cos cu succes. Sistemul reafiseaza cosul de cumparaturi cu fiecare produs, cantitate selectata si pret individual. 		
Alternative flows	8.1 Stergere multiple produse <ol style="list-style-type: none"> Administratorul selecteaza produsele din lista. Se revine la normal flow pasul 8.0.2. 		
Exceptions	8.0.E1 Stergerea s-a actionat fara selectarea unui produs <ol style="list-style-type: none"> Sistemul avertizeaza utilizatorul ca trebuie sa selecteze un produs prima data. Se revine la normal flow pasul 8.0.1. 		

CU-9: Plasare comanda produse

ID and name	CU-9: Plasare comanda produse		
Primary actor	Agentul de vanzari	Secondary actors	-
Description	<p>Agentul de vanzari lanseaza o noua comanda ce contine produsele adaugate anterior de acesta in cos impreuna cu cantitatea din fiecare produs specificata. Comanda va avea ca operator agentul de vanzari care a initiat-o. Comanda primeste statusul de PLACED. Sistemul rezerva cantitatile din fiecare produs specificate din stocul existent. Se actualizeaza stocul produselor in lista tuturor produselor.</p>		
Trigger	Agentul de vanzari da click pe butonul de plasare comanda.		
Preconditions	<p>PRE-1. Agentul de vanzari este autentificat.</p> <p>PRE-2. Contul este unul valid cu functia de agent de vanzari.</p> <p>PRE-3. Fereastra principala a agentului de vanzari este afisata.</p>		
Postconditions	<p>POST-1. Sistemul adauga o noua comanda.</p> <p>POST-2. Se actualizeaza lista comenzilor plasate de catre agentul de vanzari pentru a include noua comanda.</p> <p>POST-3. Stocul produselor selectate in comanda scade cu cantitatea specificata pentru fiecare produs in aceasta.</p>		
Normal flow	<p>9.0 Plasare comanda produse</p> <ol style="list-style-type: none"> 1. Agentul de vanzari selecteaza un produs din lista de produse pe care doreste sa il adauge in cos. 2. Agentul de vanzari selecteaza cantitatea din produs pe care doreste sa o adauge in cos. 3. Agentul de vanzari declanseaza adaugarea in cos prin apasarea butonului de adauga in cos. 4. Sistemul actualizeaza lista produselor din cos. 5. Agentul de vanzari introduce numele clientului pentru care se realizeaza comanda. 6. Agentul de vanzari plaseaza comanda prin apasarea butonului de plasare comanda. 7. Sistemul informeaza agentul de vanzari ca s-a plasat cu succes comanda. 		

	<p>8. Sistemul goleste cosul de produse.</p> <p>9. Sistemul actualizeaza lista de produse cu noul stoc al fiecarui produs dupa plasarea comenzi.</p>
Alternative flows	<p>9.1 Plasare comanda cu mai multe produse</p> <p>1. Agentul selecteaza alt produs din lista de produse dupa care se repeta pasii din normal flow 9.0.2 – 9.0.4</p> <p>2. Se revine la normal flow pasul 9.0.5</p>
Exceptions	<p>9.0.E1 Cosul de produse este gol</p> <p>1. Sistemul avertizeaza utilizatorul ca trebuie sa adauge cel putin un produs in cos inainte de a plasa comanda.</p> <p>2. Se revine la normal flow pasul 9.0.1.</p> <p>9.0.E2 Produsul selectat nu mai este in stoc</p> <p>1. Sistemul avertizeaza utilizatorul ca produsul selectat nu mai este in stoc si nu ii permite adaugarea acestuia in cos.</p> <p>2. Se revine la pasul 9.0.1.</p> <p>9.0.E3 Numele clientului nu a fost specificat</p> <p>1. Sistemul avertizeaza utilizatorul ca nu s-a putut plasa comanda deoarece numele clientului nu a fost specificat.</p> <p>2. Se revine la pasul 9.0.5.</p>

CU-10: Vizualizare comenzi plasate

ID and name	CU-10: Vizualizare comenzi plasate		
Primary actor	Agentul de vanzari	Secondary actors	-
Description	Agentului de vanzari i se prezinta lista cu toate comenzile plasate de catre acesta si anume numele clientului, stausul comenzi, precum si numarul total de produse din comanda.		
Trigger	Agentul de vanzari a dat click pe butonul de afisare a comenzilor.		
Preconditions	PRE-1. Agentul de vanzari este autentificat.		

	<p>PRE-2. Contul este unul valid cu functia de agent de vanzari.</p> <p>PRE-3. Fereastra principala a agentului de vanzari este afisata.</p>
Postconditions	POST-1. Lista cu toate comenziile plasate de catre agentul de vanzari autentificat in aplicatie este afisata.
Normal flow	<p>10.0 Vizualizare comenzi plasate</p> <ol style="list-style-type: none"> 1. Agentul de vanzari apasa butonul de vizualizare a tuturor comenziilor plasate de catre acesta. 2. Sistemul afiseaza fereastra ce contine lista cu toate comenziile plasate de catre agent.
Alternative flows	-
Exceptions	-

CU-11: Anulare comanda

ID and name	CU-11: Anulare comanda		
Primary actor	Agentul de vanzari	Secondary actors	-
Description	Agentului de vanzari selecteaza o comanda din lista de comenzi si o anuleaza, aceasta fiind stearsa definitiv.		
Trigger	Agentul de vanzari selecteaza o comanda si apasa butonul pentru stergere.		
Preconditions	<p>PRE-1. Agentul de vanzari este autentificat.</p> <p>PRE-2. Contul este unul valid cu functia de agent de vanzari.</p> <p>PRE-3. Fereastra cu lista comenziilor plasate de agentul de vanzari este afisata.</p>		
Postconditions	POST-1. Lista cu toate comenziile plasate de catre agentul de vanzari autentificat in aplicatie este actualizata.		
Normal flow	<p>11.0 Anulare comanda</p> <ol style="list-style-type: none"> 1. Agentul de vanzari selecteaza o comanda din lista de comenzi plasata de el. 		

	<p>2. Agentul de vanzari declanseaza anularea prin apasarea butonului de anulare a comenzi.</p> <p>3. Sistemul informeaza agentul ca s-a anulat cu succes comanda.</p> <p>4. Sistemul actualizeaza lista de comenzi plasate de catre agentul de vanzari curent.</p>
Alternative flows	-
Exceptions	<p>11.0.E1 Anularea s-a actionat fara selectarea unei comenzi</p> <p>1. Sistemul averteaza utilizatorul ca trebuie sa selecteze o comanda prima data.</p> <p>2. Se revine la normal flow pasul 11.0.1.</p>

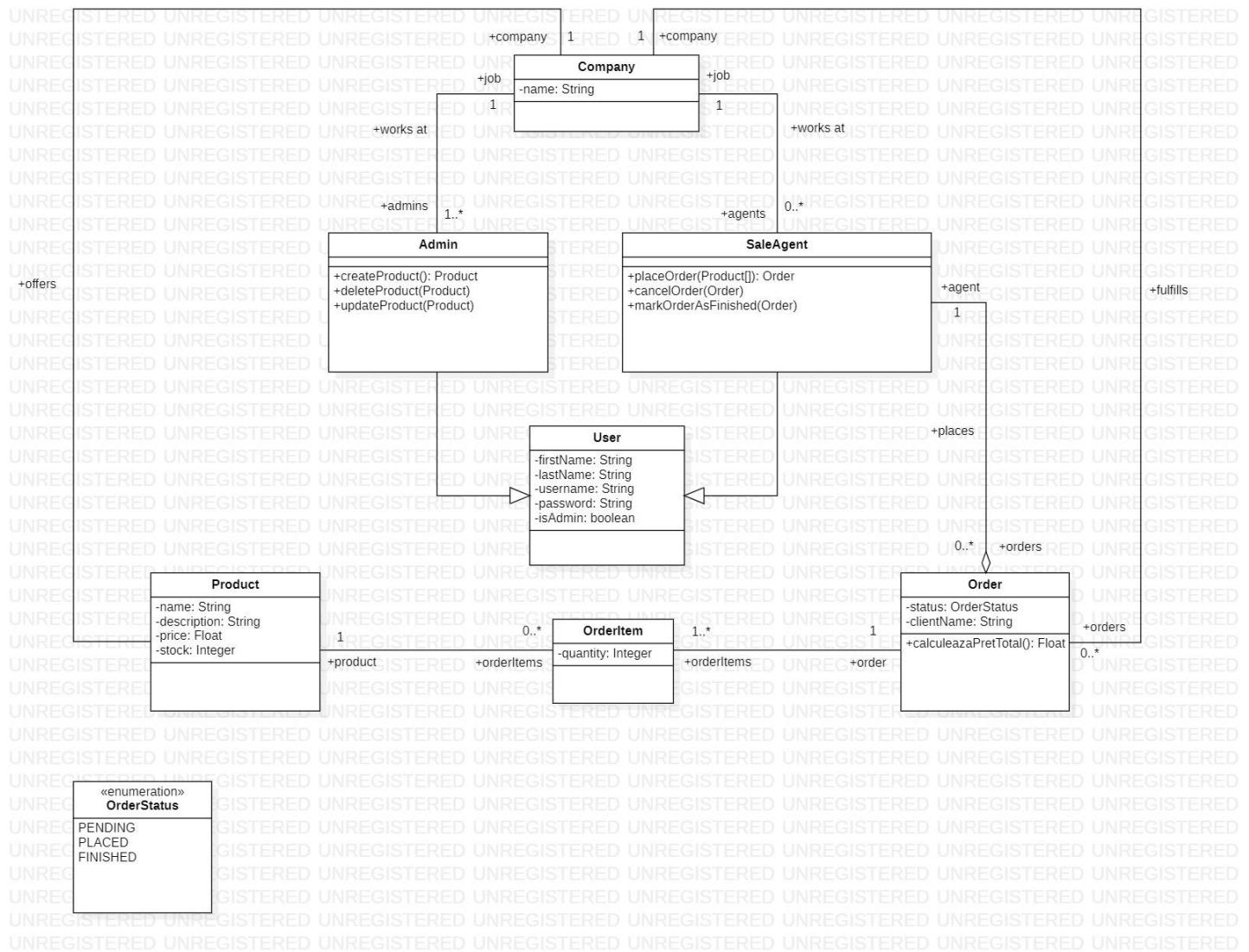
CU-12: Finalizare comanda

ID and name	CU-12: Finalizare comanda		
Primary actor	Agentul de vanzari	Secondary actors	-
Description	Agentului de vanzari selecteaza o comanda din lista de comenzi si o marcheaza ca fiind FINISHED.		
Trigger	Agentul de vanzari selecteaza o comanda si apasa butonul pentru finalizare.		
Preconditions	PRE-1. Agentul de vanzari este autentificat. PRE-2. Contul este unul valid cu functia de agent de vanzari. PRE-3. Fereastra cu lista comenzilor plasate de agentul de vanzari este afisata.		
Postconditions	POST-1. Lista cu toate comenzile plasate de catre agentul de vanzari autentificat in aplicatie este actualizata.		
Normal flow	<p>12.0 Finalizare comanda</p> <p>1. Agentul de vanzari selecteaza o comanda din lista de comenzi plasata de el.</p> <p>2. Agentul de vanzari declanseaza finalizare comenzii prin apasarea butonului de marcarea comanda ca finalizata.</p>		

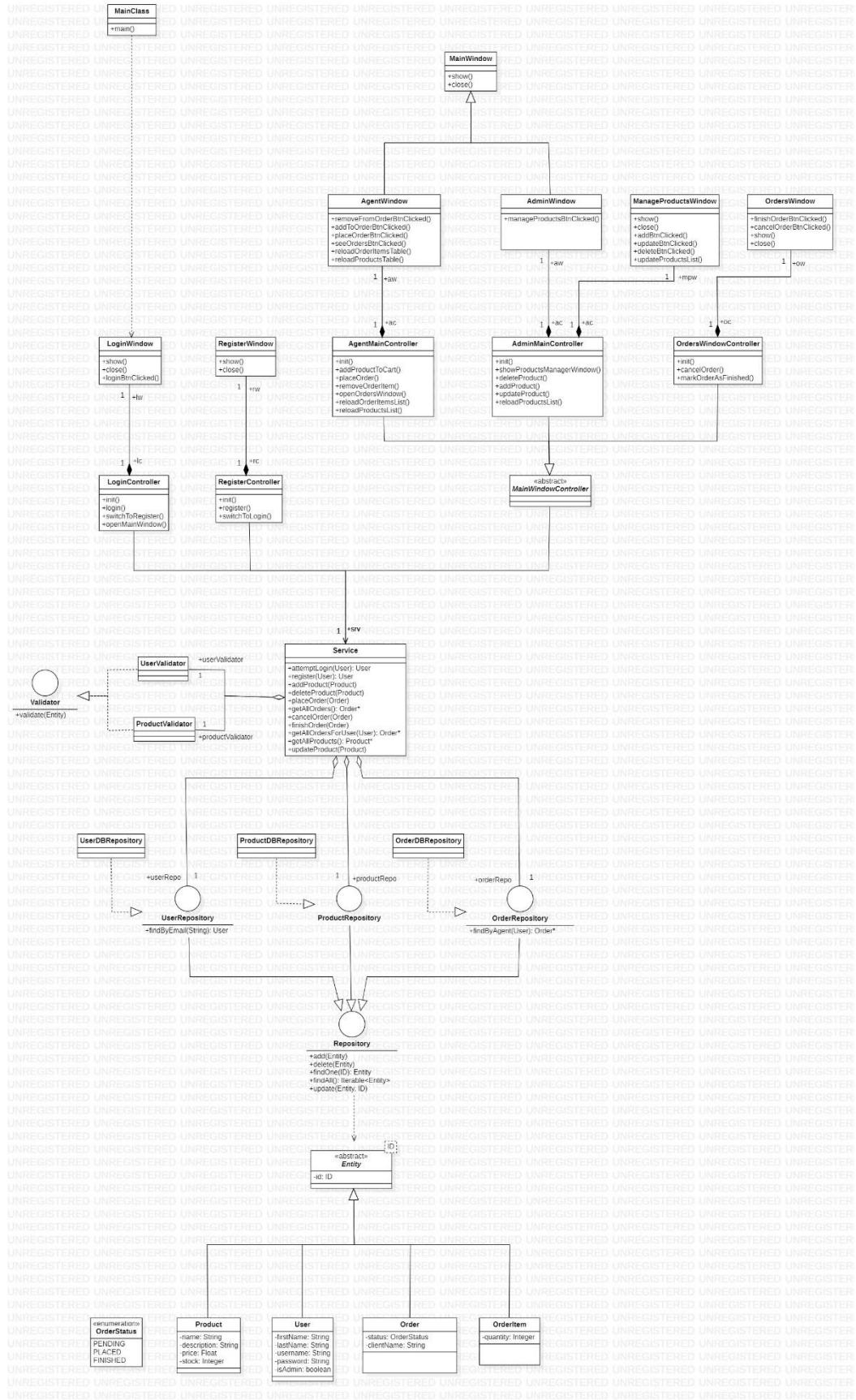
	<p>3. Sistemul informeaza agentul ca s-a finalizat comanda cu succes prin marcarea statusului acesteia la FINISHED.</p> <p>4. Sistemul actualizeaza lista de comenzi plasate de catre agentul de vanzari curent.</p>
Alternative flows	-
Exceptions	<p>12.0.E1 Finalizarea s-a actionat fara selectarea unei comenzi</p> <p>1. Sistemul avertizeaza utilizatorul ca trebuie sa selecteze o comanda prima data.</p> <p>2. Se revine la normal flow pasul 12.0.1.</p>

3. Modelul conceptual

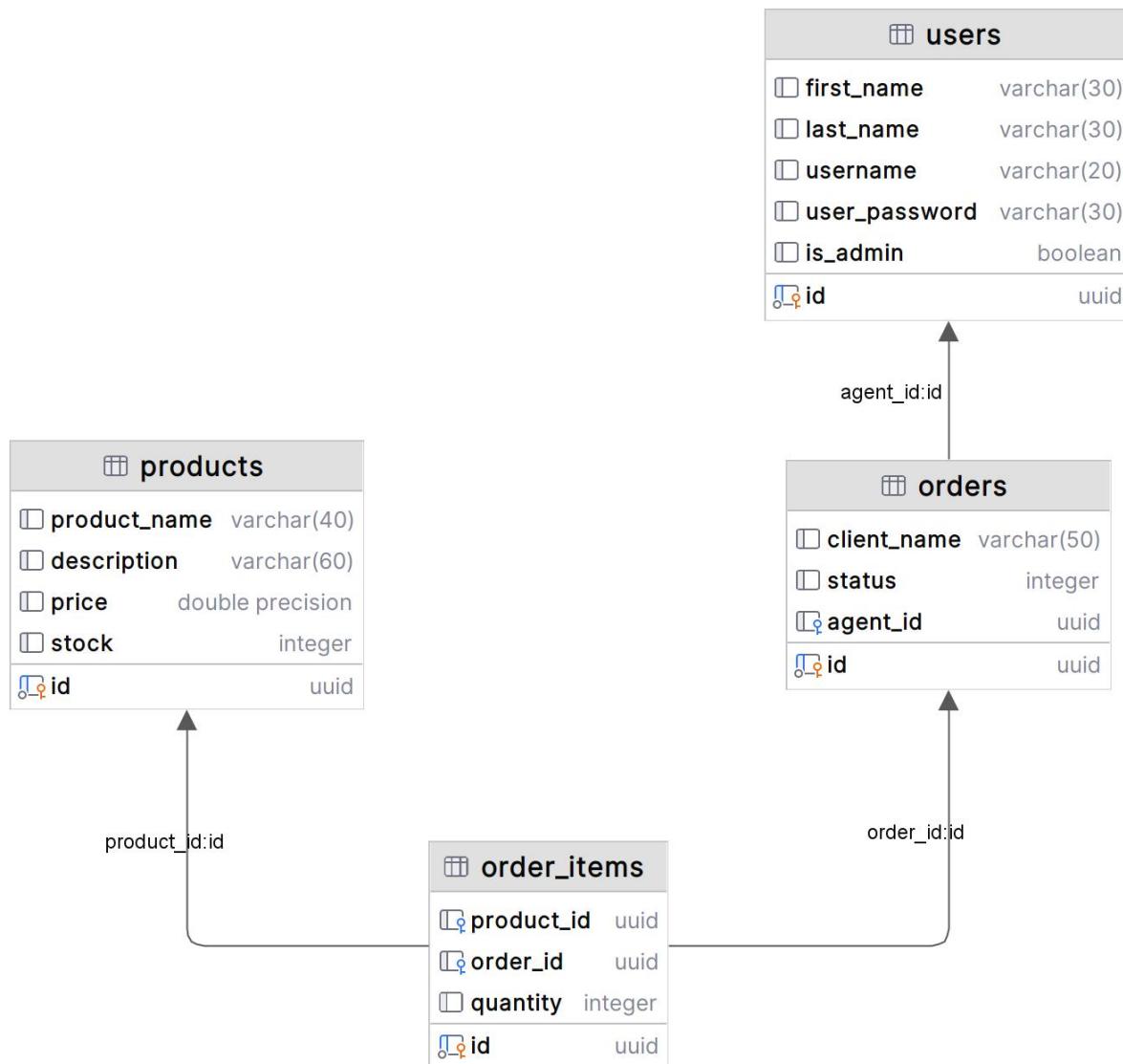
3.1 Diagrama de clase a modelului



3.2 Diagrama de clase finisata



4. Diagrama bazei de date

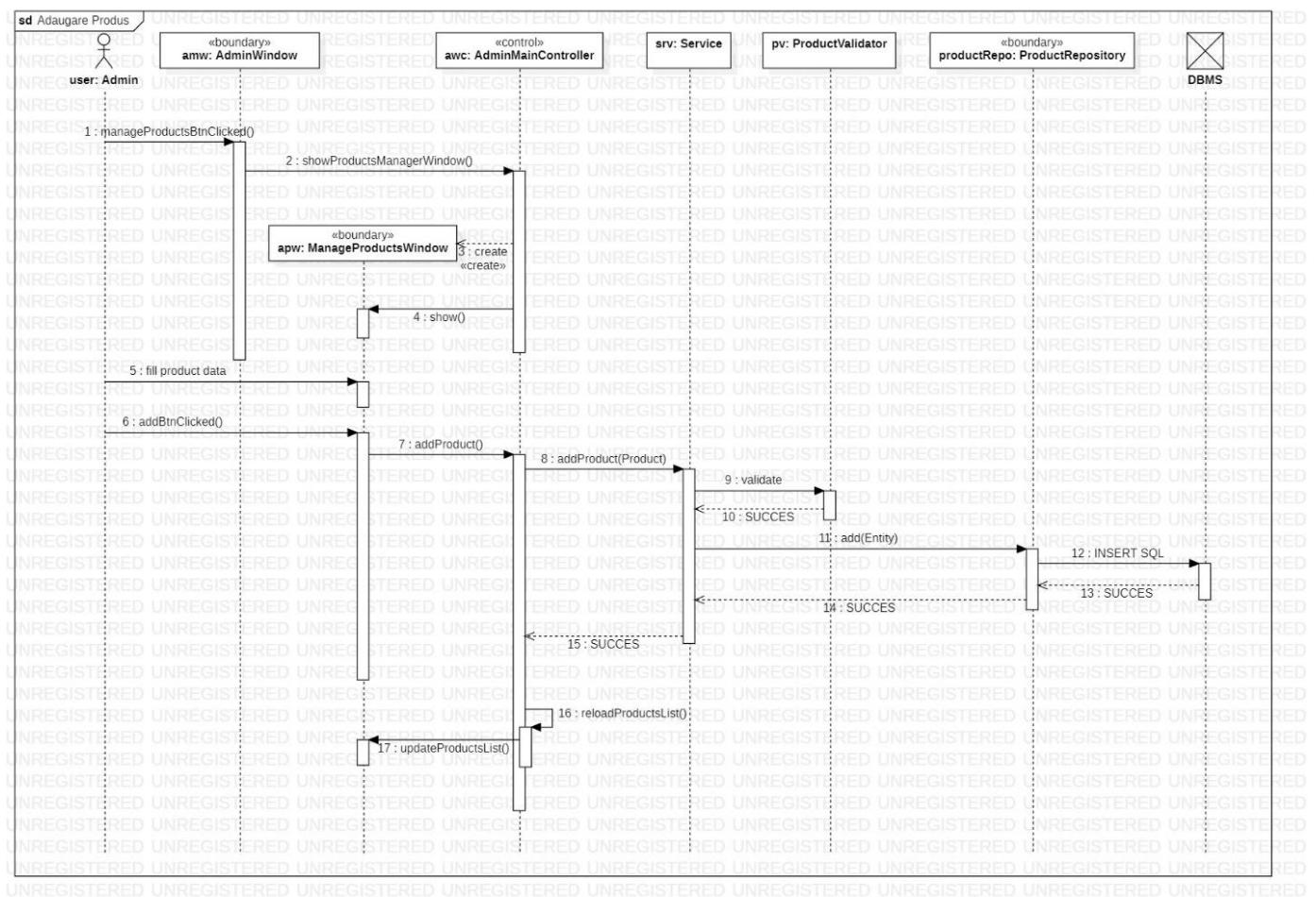


5. Modelul dinamic

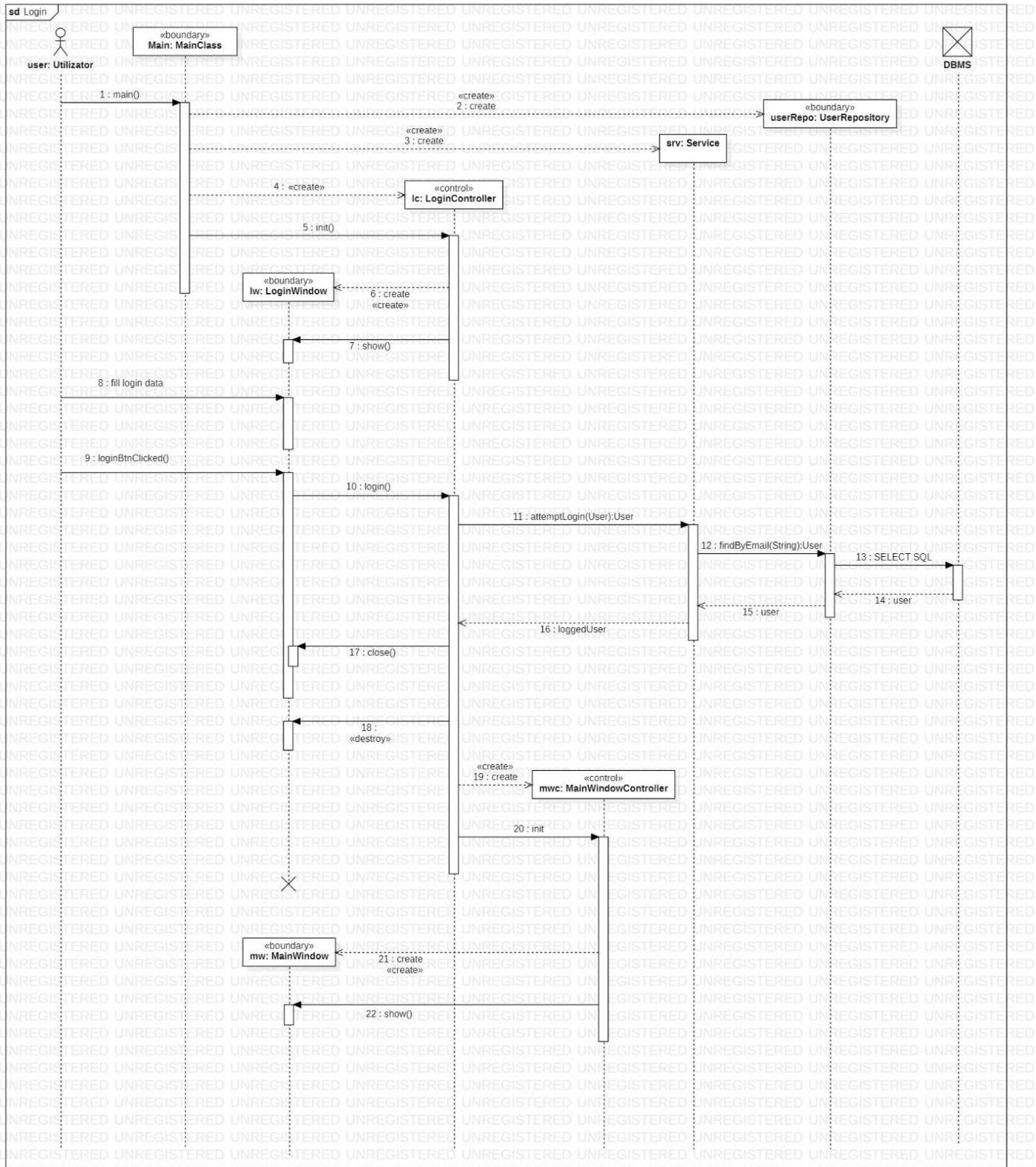
5.1 Diagrame de secvență

- Iterația 1

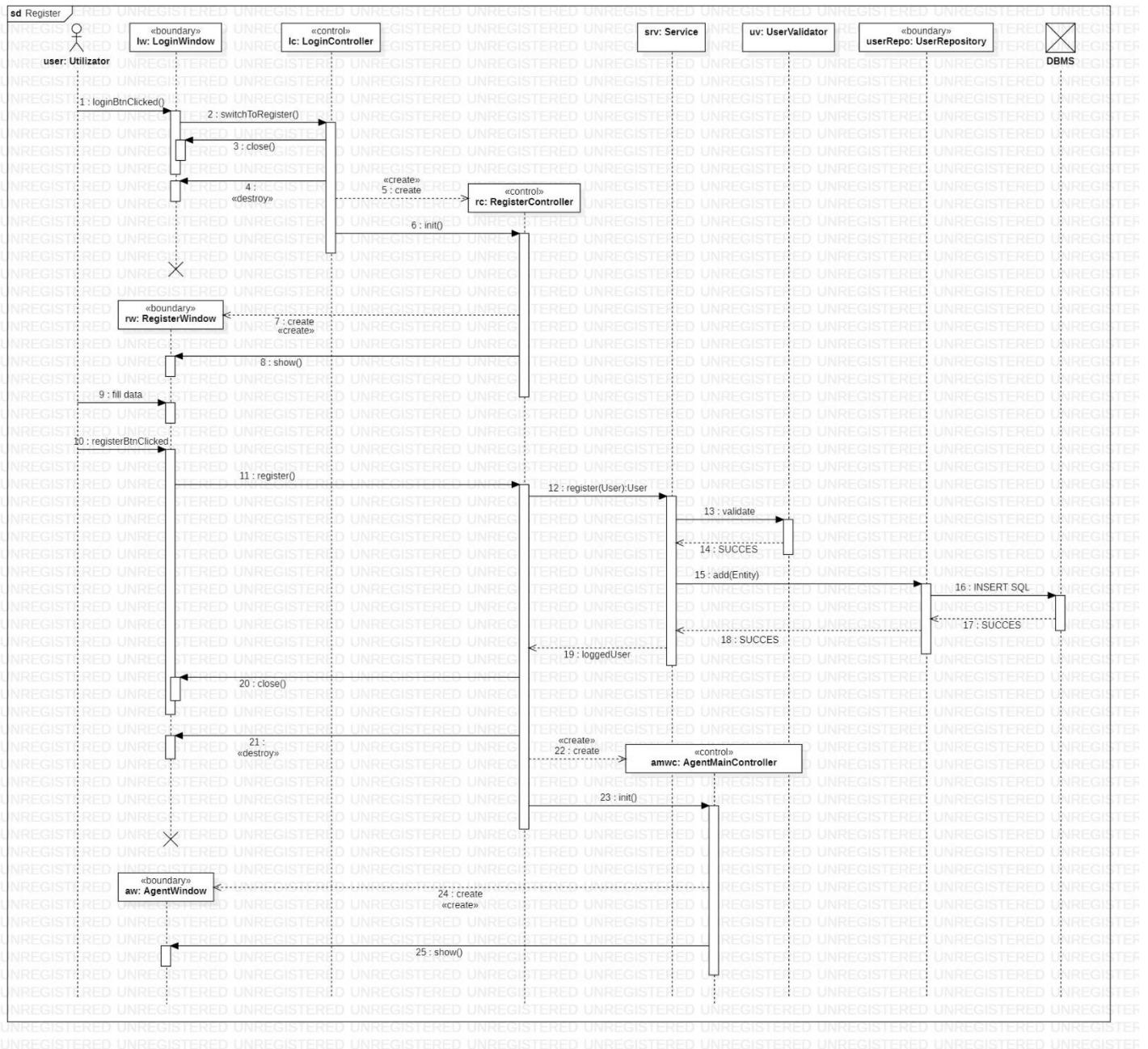
5.1.1 Adaugare Produs



5.1.2 Login

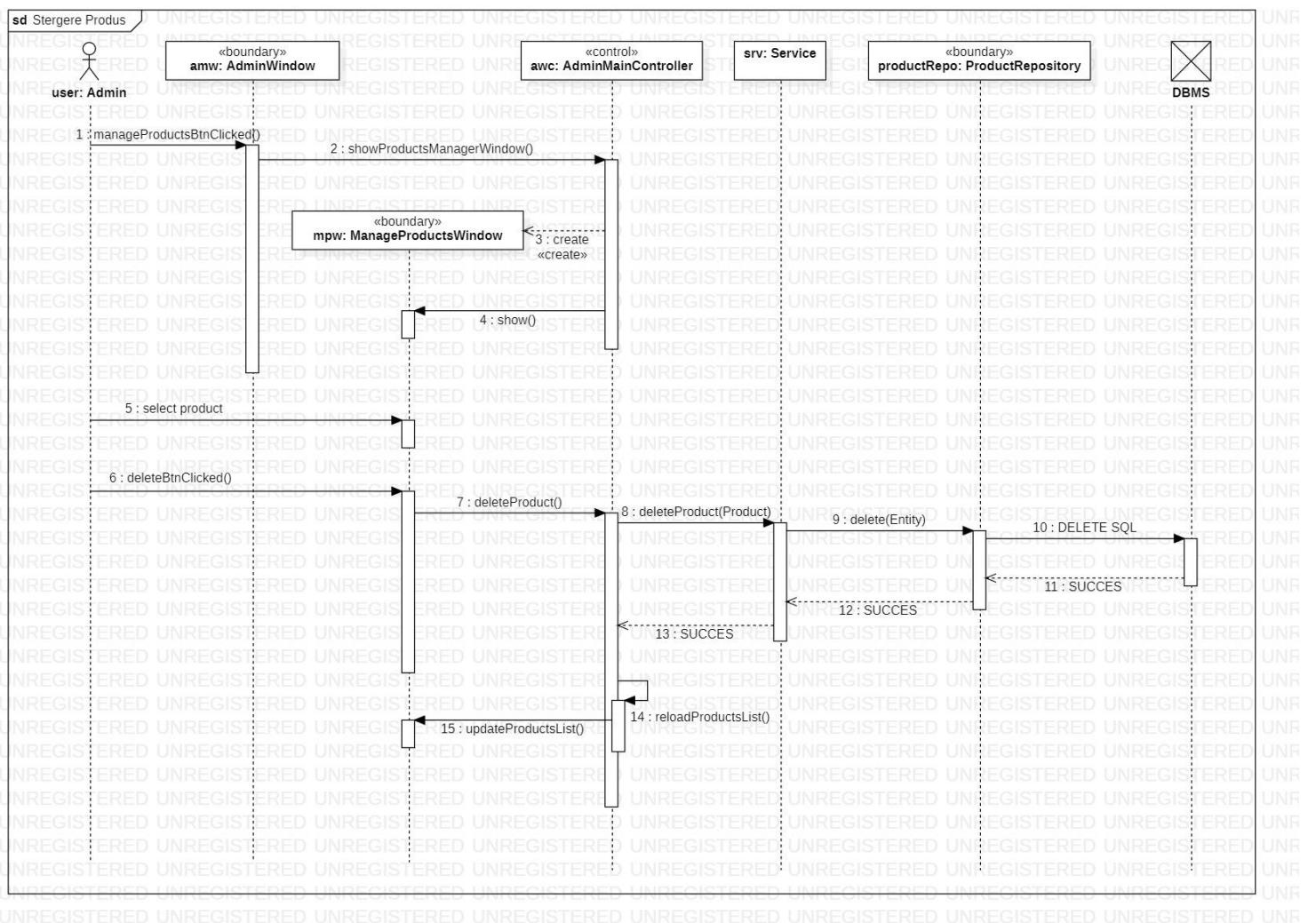


5.1.3 Register

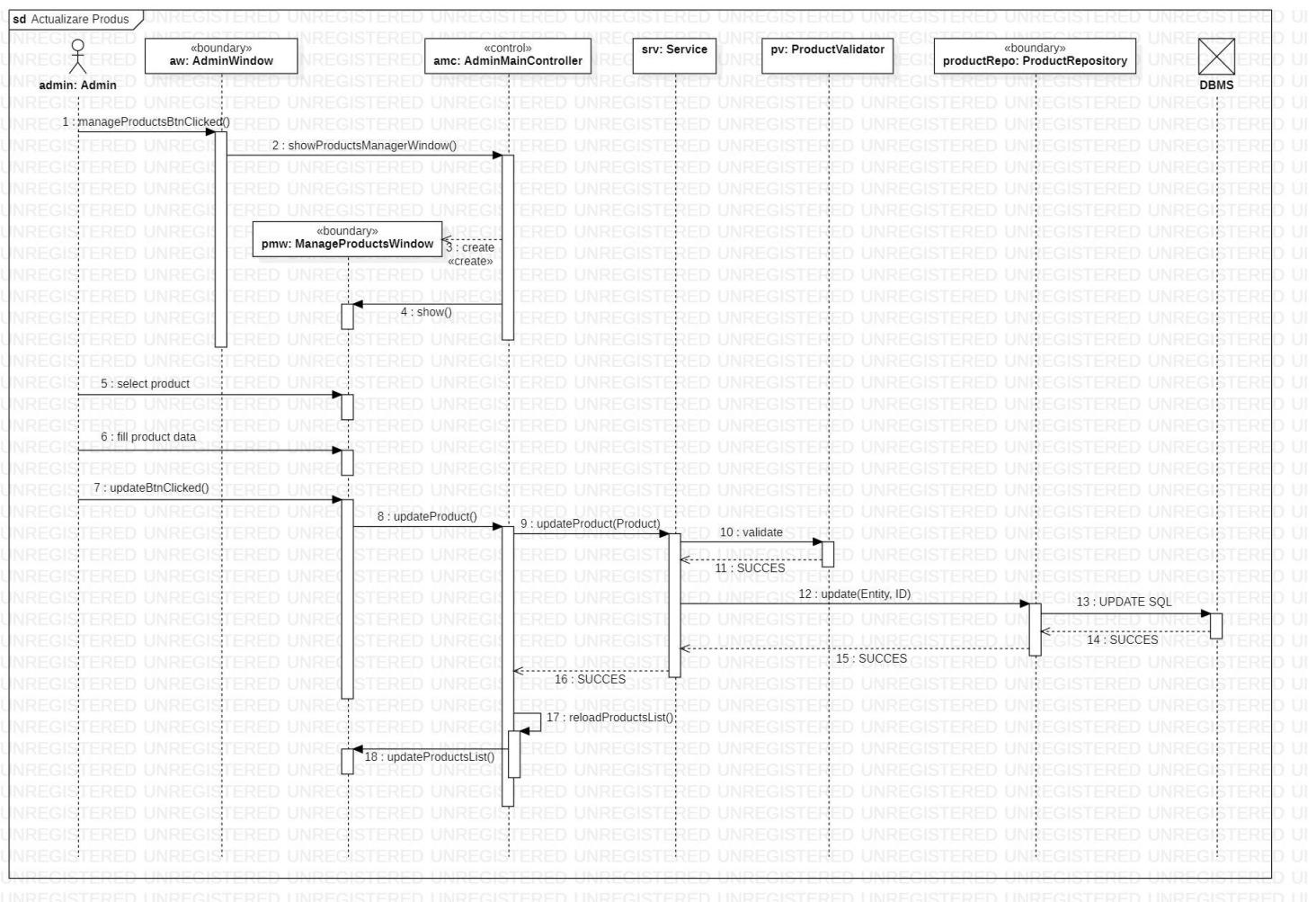


• Iteratia 2

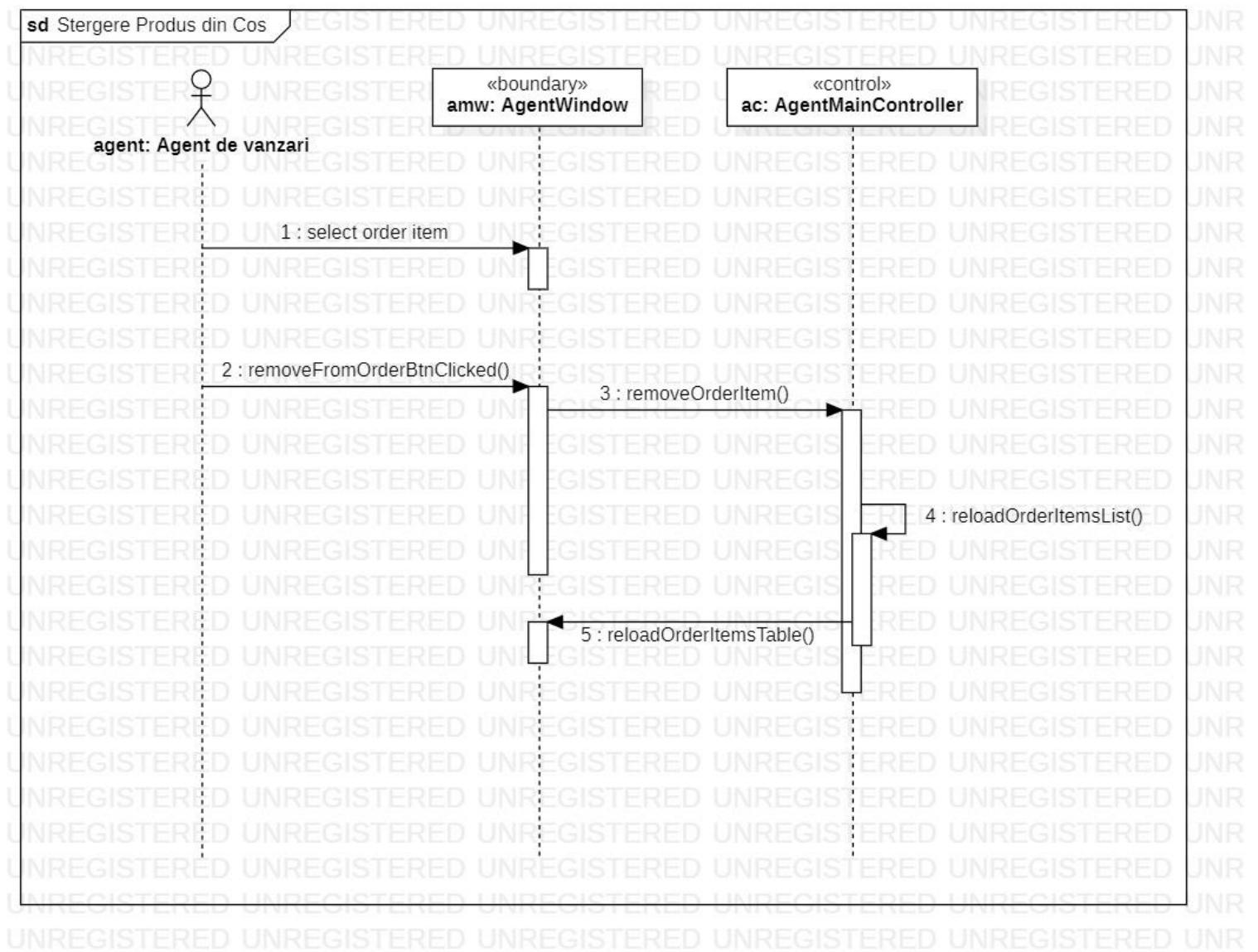
5.1.4 Stergere Produs



5.1.5 Actualizare Produs

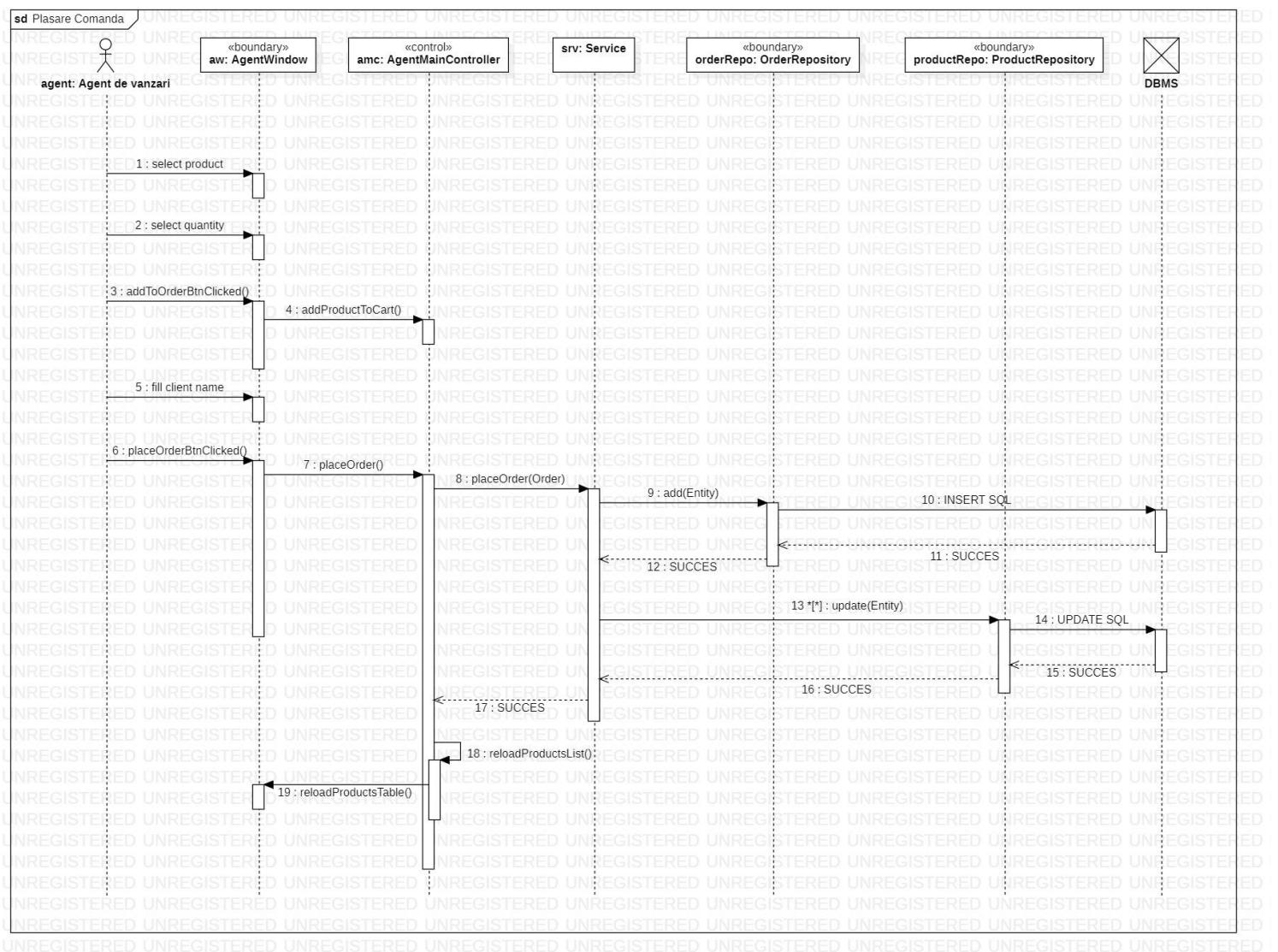


5.1.6 Stergere Produs din Cos

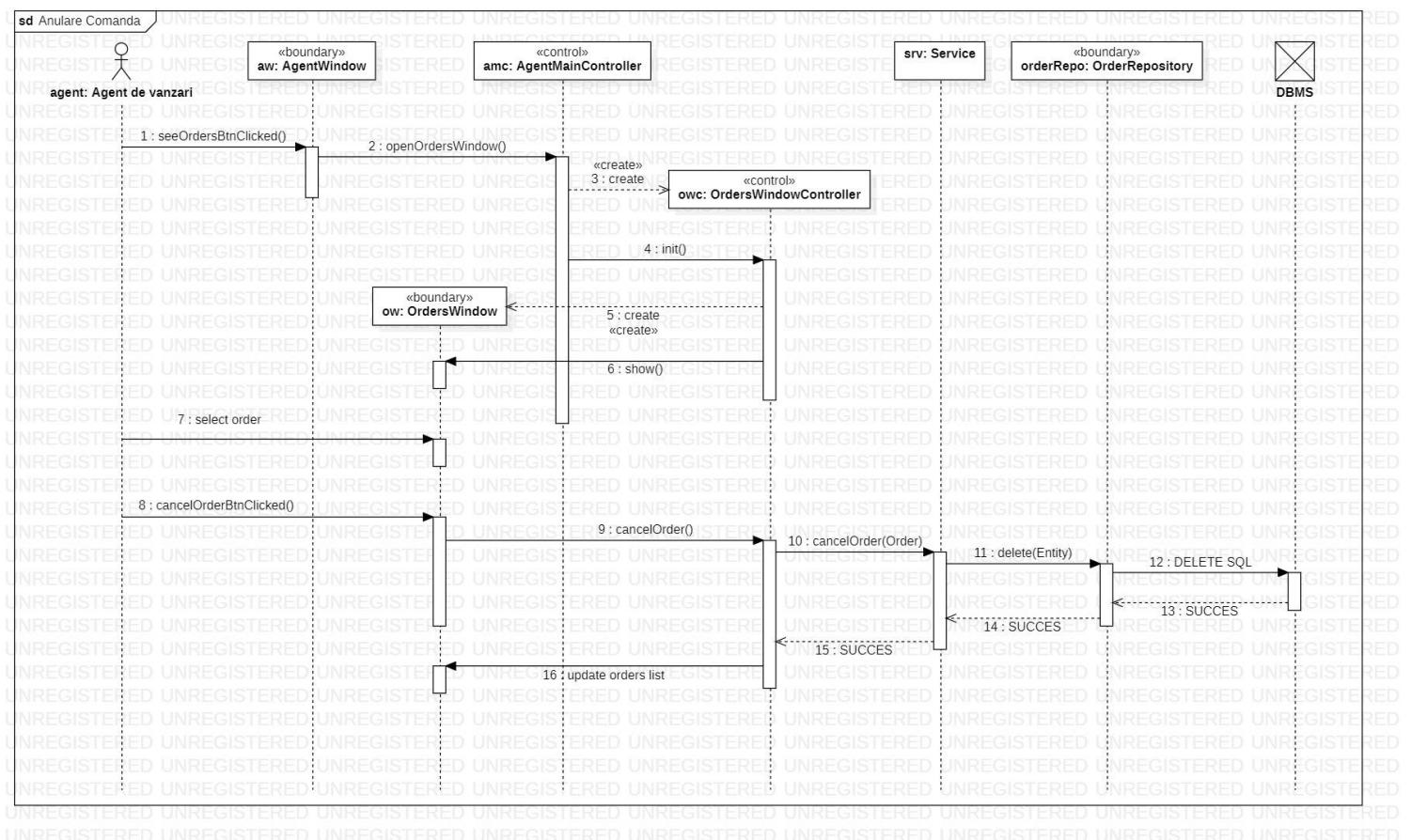


● Iteratia 3

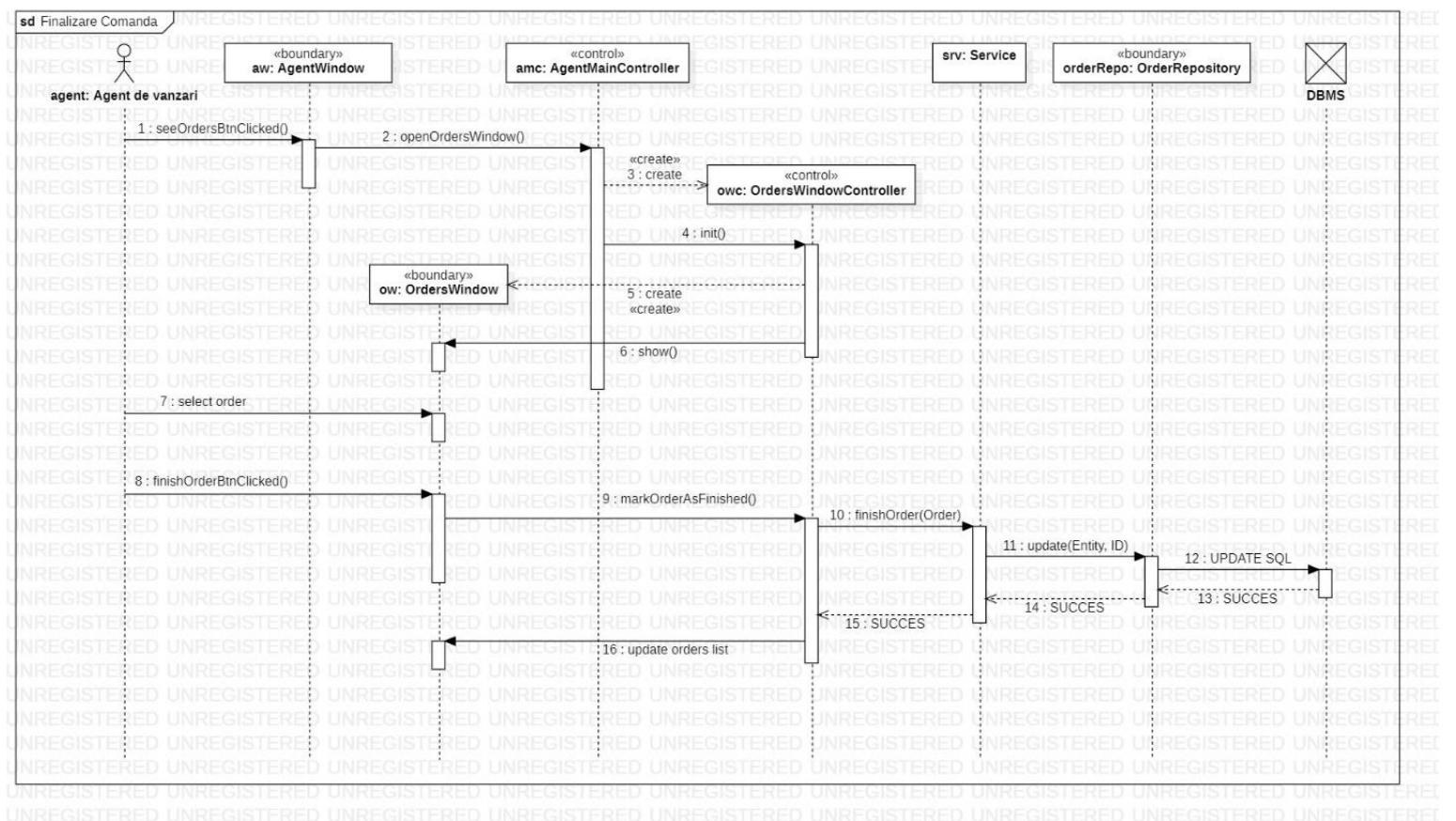
5.1.7 Plasare Comanda



5.1.8 Anulare Comanda



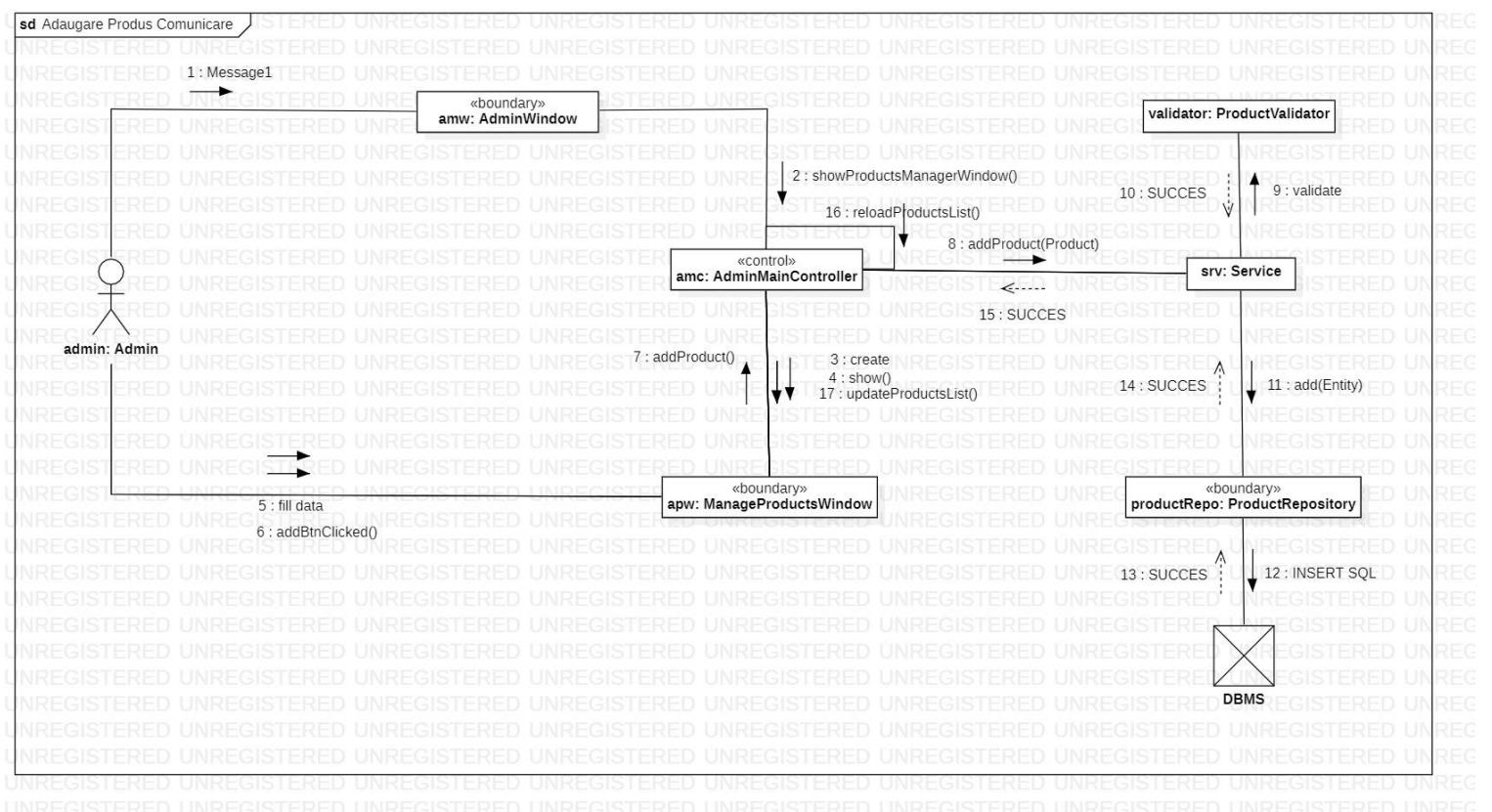
5.1.9 Finalizare Comanda



5.2 Diagrame de comunicare

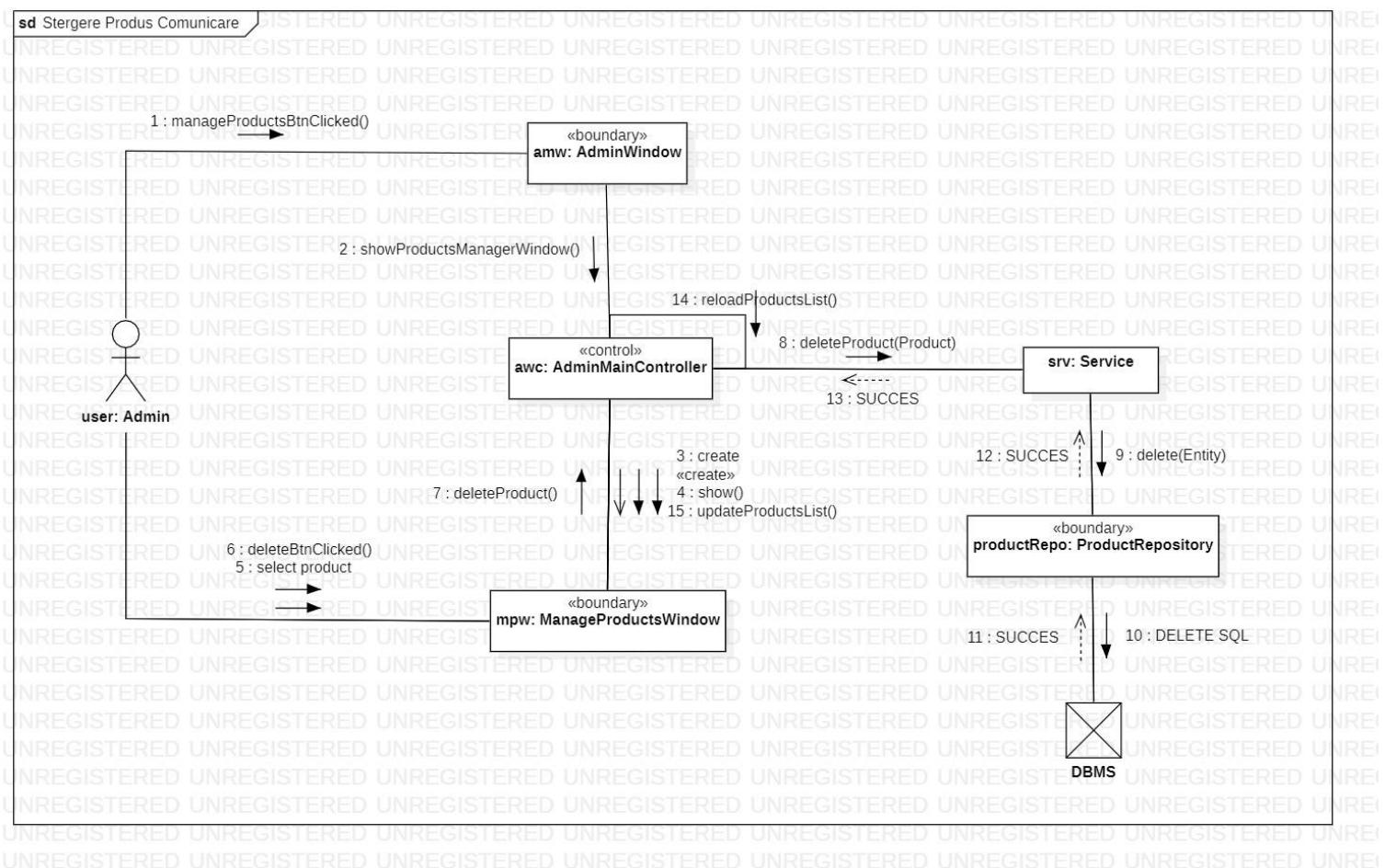
- Iteratia 1

5.2.1 Adaugare Produs



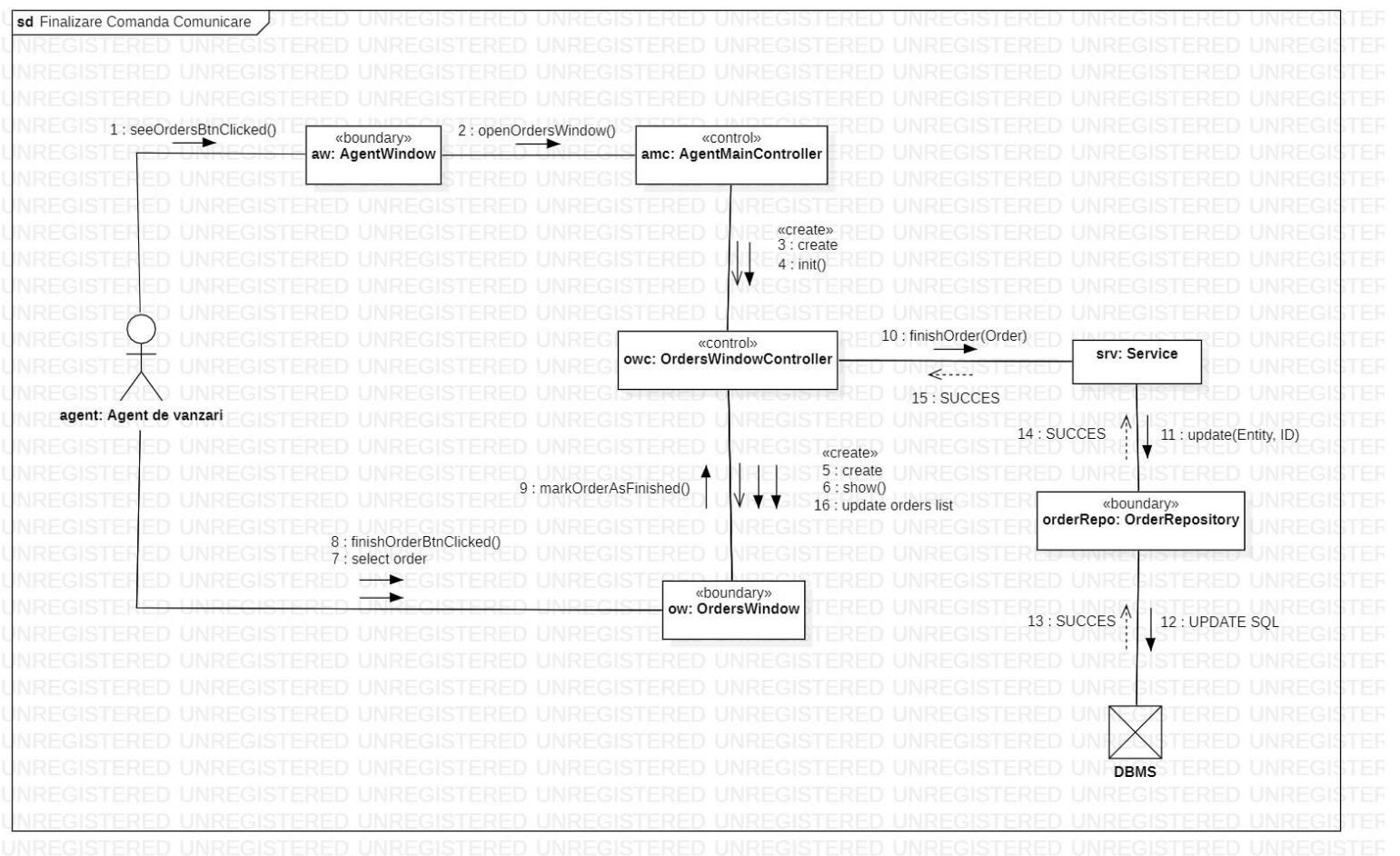
- Iteratia 2

5.2.2 Stergere Produs



● Iteratia 3

5.2.3 Finalizare Comanda



6. Documentatia tehnica

Cerinte non-functionale:

- Aplicatia trebuie sa fie una desktop
- Aplicatia trebuie sa aiba un timp de raspuns mic pentru a permite utilizatorului sa interactioneze rapid cu aceasta
- Aplicatia trebuie sa utilizeze un ORM pentru interactiunea cu baza de date indiferent de SGBD-ul utilizat
- Interfata grafica intuitiva si prietenoasa pentru a facilita utilizarea aplicatiei de catre utilizatori

Framework-uri si tool-uri:

- Ca limbaj de programare am ales Java, deoarece este un limbaj popular cu care sunt familiarizat, independent de platforma de operare si cu o comunitate de utilizatori existenta si care, de asemenea, ofera o multitudine de framework-uri.
- Pentru gestiunea dependentelor proiectului si utilizarea librariilor si framework-urilor externe am utilizat Gradle.
- Am utilizat Java Spring framework pentru configurarea aplicatiei, deoarece ofera un model de programare simplu si coerent in gestiunea dependentelor, ceea ce duce la un cod mai clar si mai usor de intretinut.
- Persistenta datelor am asigurat-o folosind PostgreSQL, un sistem de gestiune a bazelor de date robust si performant ce ofera un acces rapid si sigur la datele stocate.

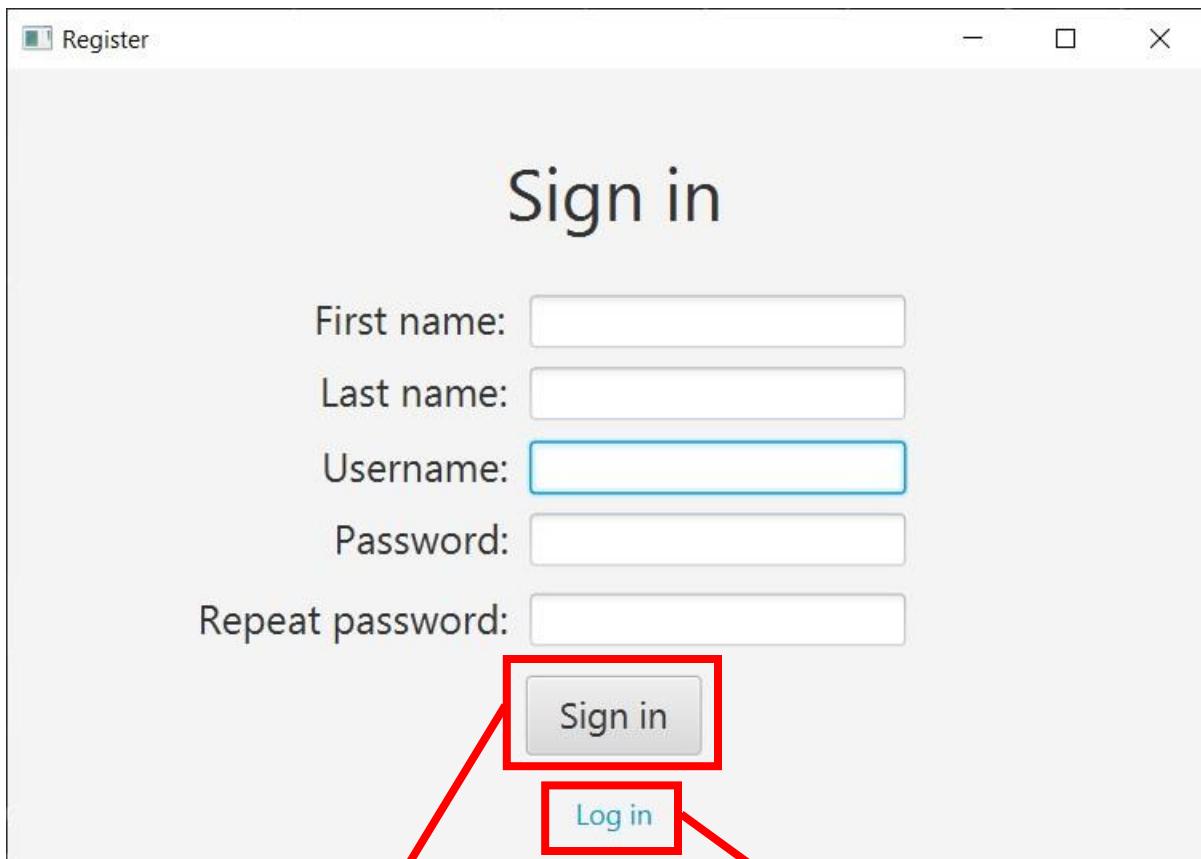
- Ca ORM am ales sa utilizez **Hibernate**, deoarece este foarte popular si usor de configurat, putand fi folosit cu o multitudine de sisteme de gestiune a bazelor de date si independent de cel ales.
- Pentru GUI am folost **JavaFX** pentru ca este un framework cu care am mai lucrat anterior si pentru ca ofera o usurinta in dezvoltarea aplicatiilor desktop. De asemenea, in combinatie cu JavaFX am folosit si **Scene Builder** pentru posibilitatea crearii de interfete grafice drag and drop.
- Pentru controlul versiunilor si salvarea progresului am folosit **Git** impreuna cu **GitHub**.
- Am folosit **StarUML** la proiectarea diagramelor pentru multitudinea de avantaje ale programului si usurinta organizarii.
- Ca mediu de dezvoltare (IDE) am ales sa folosesc **IntelliJ Ultimate**, deoarece este foarte versatil si ofera o multitudine de tool-uri si functionalitati precum gestionarea directa a bazei de date, debugging avansat, generare de diagrame, VSC etc., ce sporesc productivitatea.

7. Help

La deschiderea aplicatiei se afiseaza fereastra de login:



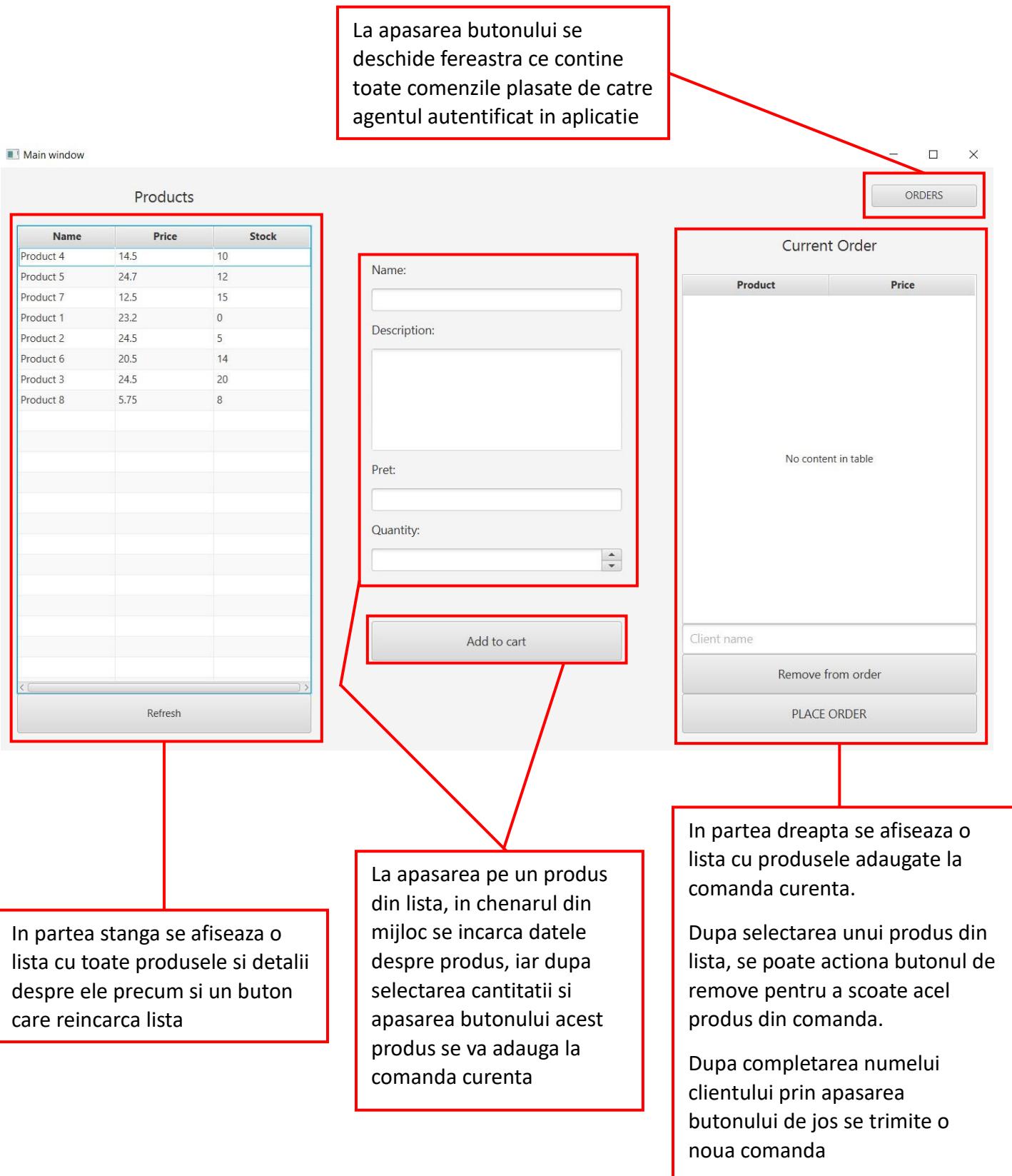
Fereastra de register:



Dupa completarea cu date a textfield-urilor, la apasarea butonului se va crea un nou cont pentru utilizator iar acesta va fi autentificat in aplicatie

In cazul in care utilizatorul are deja un cont existent la apasarea acestui buton se revine la fereastra de login

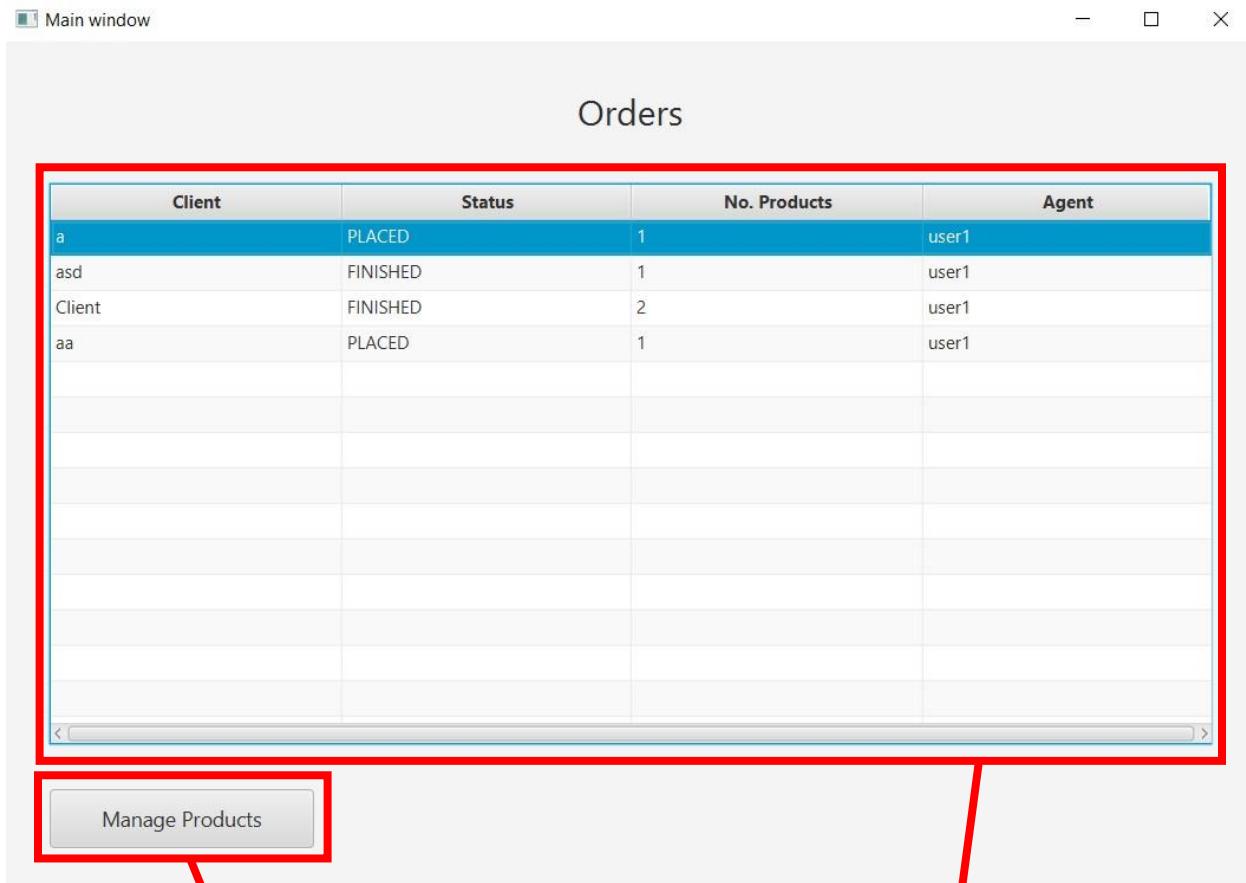
Fereastra principala pentru agentul de vanzari:



Fereastra cu comenzi agentului:



Fereastra principala pentru admin:



La apasarea butonului se deschide fereastra din care se pot aduce modificari produselor

In lista se afiseaza toate comenzile plasate de toti agentii de vanzari

Fereastra de modificare a listei de produse:

Name	Price	Stock
No content in table		

Name

Description

Price

Stock

Add

Update

Delete

Lista cu toate produsele si detalii despre fiecare

Butoane ce permit adaugarea, actualizarea si stergere produselor in functie de datele completate in textfield-urile de mai sus

8. Referinte

- Dr. Lect. Petrascu Vladiela – Ingineria Sistemelor Soft, Note de curs
- Dr. Conf. Cojocar Grigoreta – Medii de proiectare si programare, Note de curs
- <https://simacioan.wordpress.com/2023/03/13/seminarii-iss-2021/>
- <https://www.javadevjournal.com/spring-boot/spring-boot-with-hibernate/>
- <https://www.javatpoint.com/hibernate-one-to-many-mapping-using-annotation-example>
- <https://www.codejava.net/frameworks/hibernate/hibernate-parent-child-mapping-example>
- <https://www.baeldung.com/spring-boot-hibernate>
- <https://chat.openai.com/>
- <https://www.baeldung.com/spring-autowire>
- <https://stackoverflow.com/questions/3964059/jpa-default-column-name-mapping-for-manytoone-relations>
- <https://vladmirhalcea.com/manytoone-jpa-hibernate/>
- <https://vladmirhalcea.com/the-best-way-to-map-a-onetomany-association-with-jpa-and-hibernate/>