Aplicatie de tip vanzare/cumaparare – Olx

Document de analiza si design

Student: Pricop Petru Adelin

**Grupa: 30238**

**Contents**

[I Specificatia proiectului](#_heading=h.gjdgxs) 3

[1.1 Diagrama Domain Model](#_heading=h.30j0zll) 3

[II Modelul Use-Case](#_heading=h.1fob9te) 3

[2.1 Utilizatori si stakeholderi](#_heading=h.y9rd2x440h7k) 4

[2.2 Identificarea Use-Case-uri](#_heading=h.3znysh7) 5

[2.3 Diagrama UML Use-Case](#_heading=h.2et92p0) 7

[III Design arhitectural](#_heading=h.tyjcwt) 7

[3.1 Arhitectura conceptuala](#_heading=h.3dy6vkm) 7

[3.2 Diagrama de pachete](#_heading=h.1t3h5sf) 8

[3.3 Diagrama de clase](#_heading=h.4d34og8) 9

[3.4 Diagrama bazei de date](#_heading=h.2s8eyo1) 9

[3.5 Diagrame de secventa](#_heading=h.17dp8vu) 9

[3.6 Diagrame de activitati](#_heading=h.3rdcrjn) 10

[IV Specificatii suplimentare](#_heading=h.26in1rg) 10

[4.1 Specificatii non-functionale](#_heading=h.lnxbz9) 10

[4.2 Constrangeri de design](#_heading=h.35nkun2) 10

[V Testare](#_heading=h.1ksv4uv) 11

[5.1 Testarea functionalitatii aplicatiei](#_heading=h.44sinio) 11

[5.2 Dezvolatari ulterioare](#_heading=h.2jxsxqh) 11

[VI Bibliografie](#_heading=h.z337ya) 11

I Specificatia proiectului

In acest proiect se doreste implementarea unei aplicatii de tip vanzare-cumparare, asemanatoare platformei Olx, in cadrul careia utilizatorii pot posta anunturi in care prezinta produsul pe care il pun spre vanzare sau cauta anunturi in vederea achizitionarii unor bunuri sau servicii. Aplicatia este gestionata de catre administratori care pot aduce modificari asupra anunturilor, a categoriilor de produse sau a utilizatorilor.

1.1 Diagrama Domain Model

A computer screen shot of a diagram

Description automatically generated

II Modelul Use-Case

In cadrul acestui capitol vom prezenta cateva modele use-case, care vor ghida dezvolatrea si testarea, asigurand ca toti utilizatorii si stakeholderii interactioneaza efficient cu sistemul.

In cadrul aplicatiei prezentate, stakeholderii sunt reprezentati prin administratorii aplicatiei care au responsabilitatea de a gestiona utilizatorii, anunturile postate si categoriile de anunturi existente. De asemenea acestia pot genera rapoarte lunare cu numarul de anunturi care au fost postate sau pot trimite email-uri catre anumiti useri selectati.

Utilizatorii aplicatiei sunt cei care postaeaza anuturi cu intentia de a-si vinde produse sau cauta anunturi in vederea achizitionatrii unora de la alti utilizatori. Userii au posibilitatea de a se inregistra in aplicatie, de a vizualiza anunturi, de a-si gestiona anunturilesi de a trimtie mesaje unor alti utilizatori.

2.1 Utilizatori si stakeholder

|  |  |  |
| --- | --- | --- |
| Nume | User(Utilizator) | Stakeholder |
| **Admin(Administrator)** |
| Descriere | Reprezinta utilizatorii inregistrati in aplicatie care pot sa cumpere sau sa vanda prin intermediul aplicatiei create. | Reprezinta administratorii aplicatiei care se oupa de gestionarea datelor din aplicatie(ex. Useri, Anunturi, Categorii). |
| Responsabilitati | * Inregistrarea in aplicatie; * Vizualizarea anunturilor; * Filtrare anunturi in functie de categorie; * Trimitere mesaj catre un alt user pentru o postare curenta; * Postare anunt; * Editare anunt; * Stergere anunt. | * CRUD useri; * CRUD anunturi; * CRUD categorii; * Trimitere email catre un anumit user selectat; * Raport pe fiecare luna pentru cate anunturi au fost postate, in format CSV. |

2.2 Identificarea Use-Case-uri

1. ***Nume Use case****: Adaugare User*

***Nivel:*** *Subfunction****.***

***Actor principal:*** *Administratorul* ***.***

***Scenariul principal de success:***

* *Administratorul se logheaza si acceseaza pagina de administrare useri;*
* *Alege optinea de adaugare User;*
* *Introduce detaliile user-ului si apasa butonul Submit;*
* *Aplicatia afiseaza un mesaj de success cu id-ul noului user creat.*

***Extensie:***

* + - *Daca datele solicitate de aplicatie nu au fost introduce correct sau complet de catre administrator, aplicatia va afisa un mesaj de eroare.*

1. ***Nume Use case****: Stergere Categorie*

***Nivel:*** *Subfunction****.***

***Actor principal:*** *Administratorul* ***.***

***Scenariul principal de success:***

* *Administratorul se logheaza si acceseaza pagina de administrare categorii;*
* *Alege optinea de stergere categorie;*
* *Introduce id-ul categoriei si apasa butonul Delete.*
* *Aplicatia sterge categoria din baza de date si toate anunturile care fac parte din acea categorie;*
* *Aplicatia afiseaza un mesaj de success cu id-ul categoriei sterse.*

***Extensie:***

* + - *Daca nu exista o categorie cu id-ul introdus de administrator, aplicatia afiseaza un mesaj de eroare.*

1. ***Nume Use case****: Adaugare Anunt*

***Nivel:*** *User-Goal****.***

***Actor principal:*** *User-ul* ***.***

***Scenariul principal de success:***

* *User-ul se logheaza si acceseaza pagina Anunturile Mele;*
* *Alege optinea de adaugare Anunt;*
* *Introduce detaliile anuntului si apasa butonul Submit.*
* *Aplicatia afiseaza un mesaj de success cu id-ul noului anunt creat.*

***Extensie:***

* + - *Daca datele solicitate de aplicatie nu au fost introduce correct sau complet de catre user, aplicatia va afisa un mesaj de eroare si anuntul nu va fi adaugat.*

1. ***Nume Use case****: Modificare Anunt*

***Nivel:*** *User-Goal****.***

***Actor principal:*** *User-ul* ***.***

***Scenariul principal de success:***

* *User-ul se logheaza si acceseaza pagina Anunturile Mele;*
* *Alege optinea de modificare anunt;*
* *Modifica detaliile deja existente si apasa butonul Submit.*
* *Aplicatia afiseaza un mesaj de success cu id-ul anuntului modificat.*

***Extensie:***

* + - *Daca datele modificate sunt incomplete sau incorecte, aplicatia va afisa un mesaj de eroare.*

A diagram of a person with text

Description automatically generated2.3 Diagrama UML Use-Case

A diagram of a person with text

Description automatically generated

**Add/Remove announcemnt from favourite**

III Design arhitectural

În acest capitol, vom explora în detaliu fiecare componentă a arhitecturii aplicaitei noastre, astfel putandu-se observa cum contribuie fiecarecomponenta la funcționalitatea generală a aplicației, de la interfața cu utilizatorul până la gestionarea datelor.

3.1 Arhitectura conceptuala

A diagram of a company

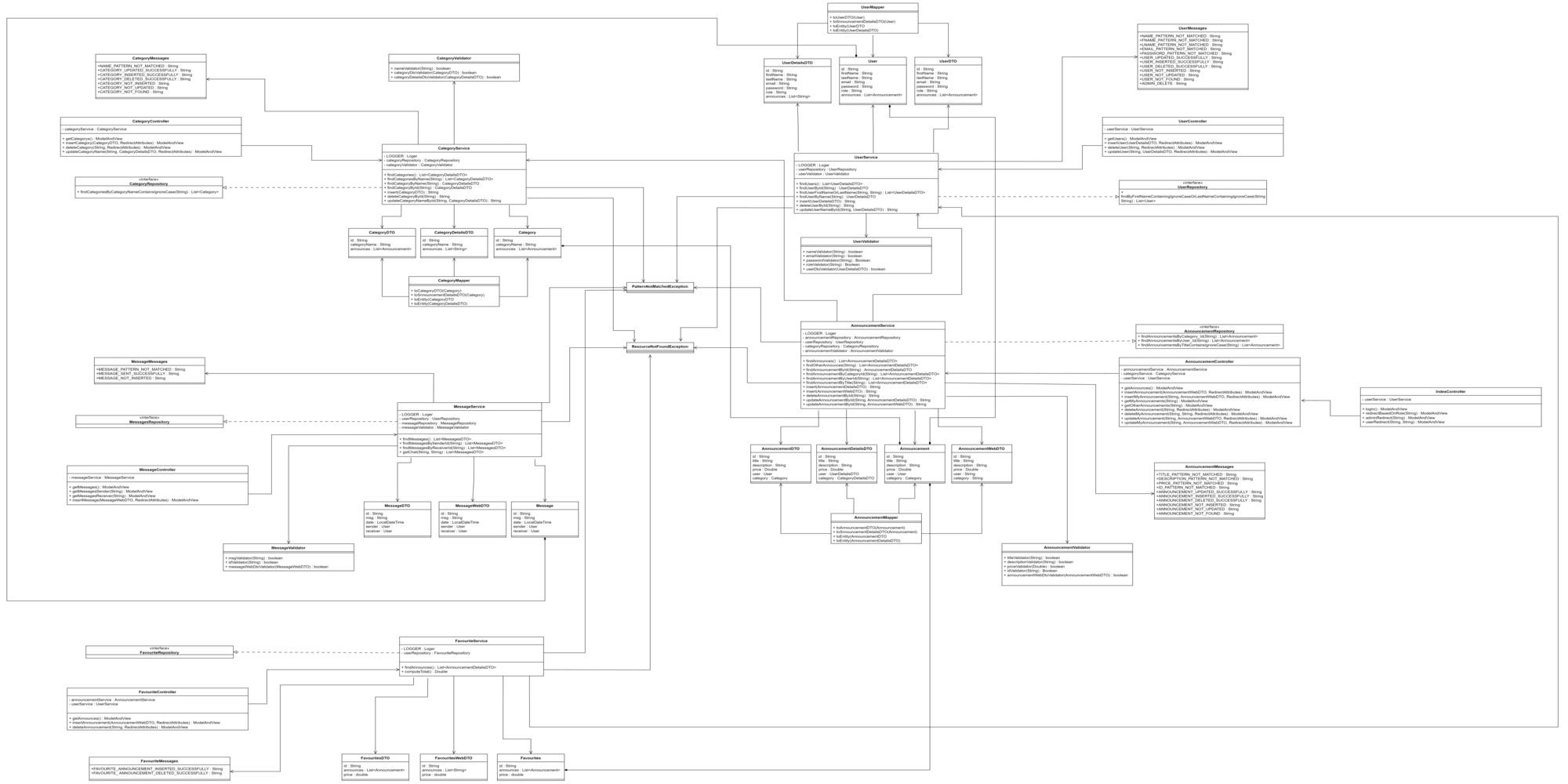
Description automatically generated

3.2 Diagrama de pachete

A diagram of a service

Description automatically generated

3.3 Diagrama de clase



3.4 Diagrama bazei de date

A diagram of a diagram

Description automatically generated with medium confidence

3.5 Diagrame de secventa

A diagram of a diagram

Description automatically generated with medium confidence

A diagram of a graph

Description automatically generated

3.6 Diagrame de activitati

A diagram of a computer

Description automatically generated

IV Specificatii suplimentare

În cadrul acestui capitol, se vor prezenta cerintele suplimentare care nu se incadreaza in categoria cerintelor functionale sau de utilizare ale aplicatiei. Aceste specificatii sunt esentiale pentru functionarea eficienta a aplicatiei, putand include cerinte de performanta, securitate, conformitate si altele.

4.1 Specificatii non-functionale

Specificatiile non-functionale ale aplicatiei sunt:

* Scalabilitatea: Aplicatia se poate adapta astefel incat sa fie capabila sa gestioneze un numar crescut de utilizatori si un volum de date mai mare.
* Utilizabilitatea: Aplicatia are un design intuitiv, usor de folosit atat de utilizatori, cat si de administratori.
* Intretinearea: Aplicatia este usor de intretinut deoarece urmeaza un design pattern bine divizat pe pachete si pe responsabilitati, iar numele claselor/pachetelor/variabilelor este unul clar si specific.

4.2 Constrangeri de design

Constrngerile de design impuse la crearea aplicatiei sunt:

* Utilizarea limbajului Java pentru partea de backend si a limbajelor HTML, JavaScript si CSS pentru frontend.
* Trimiterea de mesaje intre aplicatii cu ajutorul RabbitMQ.
* Utilizarea unei baze de date relationale dintr-un server PostrgeSQL.

V Testare

5.1 Testarea functionalitatii aplicatiei

In cadrul acestui sub-capitol se va prezenta functionarea si testarea celor doua scenarii prezentate in diagramele de secvente.

1. Adaugarea unui anunt in lista de favorite a utilizatorului:
   1. Utiilizatorul isi introduce datele de logare in aplicatie:

A screenshot of a computer

Description automatically generated

In caz de succes, utilizatorul va ajunge la urmatorul pas, altfel el va fi redirectionat tot la aceasta pagina.

* 1. Pagina deschisa reprezinta listarea anunturilor celorlaltor utilizatori. Fiecare anunt are un buton „Add to favs” care, adauga anuntul in lista de anunturi favorite ale utilizatorului logat.

A screenshot of a computer

Description automatically generated

* 1. Dupa apasarea butonului utilizatorul este redirectionat la pagina de favorite unde este afisat si pretul total al produselor din lista:

A screenshot of a computer

Description automatically generated

In cazul in care anuntul a fost deja adaugat in lista de favorite, in partea de jos a paginii va aparea un scurt mesaj de informare care ne spune ca acel anunt exista deja in lista de faorite.

1. Trimiterea unui mesaj catre un utilizator(din lista de anunturi postate)
   1. Utiilizatorul isi introduce datele de logare in aplicatie:

A screenshot of a computer

Description automatically generated

In caz de succes, utilizatorul va ajunge la urmatorul pas, altfel el va fi redirectionat tot la aceasta pagina.

* 1. Pagina deschisa reprezinta listarea anunturilor celorlaltor utilizatori. Fiecare anunt are un buton „Send Message” care deschide o ferestra in care se introduce mesajul.

A screenshot of a computer

Description automatically generated

* 1. Utilizatorul introduce mesajul in caseta de input si apasa butonul „Send”

A screenshot of a computer

Description automatically generated

* 1. Dupa apasarea butonnului, mesajul este trimis si utilizatorul este redirectionat la chatul cu vanzatorul.

A screenshot of a computer

Description automatically generated

5.2 Dezvolatari ulterioare

O posibila dezvoltare ulterioara ar putea fi un sistem de acordare de note ca grad de incredere acordat vanzatorilor pentru a reduce sansa de a fi inselat in cadrul platformei; astefel, un vanzator care a facut la un moment dat o inselaciune sa fie mai usor de identificat de catre ceilalti utilizatori.

O alta dezvolare ulterioara este reprezentata de integrarea aplicatiei cu retelele sociale pentru a fi mai usor de distribuit anunturi pe retele sociale ca sa fie mai vizibile.

Totodata, pentru a oferi o experienta mai placuta utilizatorului, se poate integra un sistem de recomandari personalizate bazate pe cautarile anterioare ale acestuia.

VI Bibliografie

[1] W3 Schools 🡪 „How TO - CSS/JS Modal”

[Online]. Availavle: <https://www.w3schools.com/howto/howto_css_modals.asp> [Accesed 26.03.2024]

[2] Spring Cloud 🡪 „Messaging with RabbitMQ in Spring Boot Application”

[Online]. Available: <https://www.springcloud.io/post/2022-03/messaging-using-rabbitmq-in-spring-boot-application/#gsc.tab=0> [Accessed 17.04.2024]

[3] Spring Boot TUTORIAL 🡪 „Spring Boot - Send HTML & Text E-Mail with Thymeleaf Templates part 03” [Online]. Available: <https://www.youtube.com/watch?v=Sst9O5C6WhQ&ab_channel=SpringBootTUTORIAL>. [Accesed 18.04.2024]

[4] W3 Schools 🡪 „HTML Tutorial”

[Online]. Available: <https://www.w3schools.com/html/> [Accessed 25.03.2024]