**ACADEMIA DE STUDII ECONOMICE DIN BUCUREȘTI, FACULTATEA DE CIBERNETICĂ, STATISTICĂ ȘI INFORMATICĂ ECONOMICĂ**



**PROIECT PPOO**

Tema 3 – Gestiunea unui magazin online

Profesor Coordonator: Ciurea Eugen-Cristian

Student: Marin Daniel-Constantin

**București, 2022**

Cuprins

[Introducere 3](#_Toc119103040)

[Descrierea claselor 3](#_Toc119103041)

[Descrierea aplicației 4](#_Toc119103042)

[Concluzie 4](#_Toc119103043)

[Bibliografie 5](#_Toc119103044)

# Introducere

Tema proiectului a fost aceea de a crea o aplicație de gestiune a unui magazin online de produse. Pentru proiectul acesta am ales să gestionez un magazin online de telefoane mobile și care se ocupă de vânzarea și cumpărarea de diverse telefoane mobile. Toate produsele sunt salvate în fișiere text. Există două tipuri de persoane care pot interacționa cu aplicația aceasta, iar aceștia sunt utilizatorii și administratorii.

Aplicația are ca scop prin meniul interactiv să permită cu ușurință cumpărarea unui telefon mobil, să poată filtra în funcție de anumite criterii și să poată urmării ce telefoane au mai fost cumpărate până în momentul respectiv.

# Descrierea claselor

Aplicația este alcătuită din 15 clase, dintre care există o excepție creată de programator, dar și 4 fișiere text în care sunt salvate și manipulate toate datele despre aplicație și utilizatori. Clasele de Admin si User au aceeași componență, acestea având fiecare câte 3 atribute, acestea fiind name, password și birthDate. Clasa Phone conține mai multe atribute, acestea fiind id, index, name, model, size, year, color, basePrice, isBought.

Plecând de la clasa Phone, există încă 4 clase care extind clasa Phone, acestea fiind clasele IPhone, Xiaomi, Samsung și Huawei, având aceleași atribute, diferența o face numele prestabilit al telefonului. Pe lângă aceste clase mai există 2 clase care se leagă de câteva caracteristici ale produselor aceste clase fiind clasa Features care conține o metodă prin care pot să plătesc în rate produsul achiziționat, dar și clasa FilterService care conține mai multe metode prin care se pot filtra produsele în funcție de anumite criterii.

Pe lângă clasele existente, am creat o excepție care avertizează utilizatorul dacă numele ales mai există deja în listă, făcând astfel o listă unică de utilizatori, dar și un Enum prin care există o enumerație de culori pentru telefoane.

Ultimele 3 clase care fac posibilă funcționarea aplicației sunt clasa Shop în care sunt create toate metodele și meniurile interactive ale aplicației, clasa Main care apelează o metodă de start() a aplicației, dar și o clasa DataSource care face conexiunea și preia din fișiere și salvează în liste toate produsele și datele existente.

# Descrierea aplicației

La pornirea aplicației utilizatorul sau administratorul este întâmpinat de un meniu interactiv în care se poate autentifica, poate crea un cont nou, poate viziona telefoanele fără a fi autentificat. În momentul în care clientul se autentifică în aplicație, acesta este întâmpinat de un alt meniu din care poate să aleagă mai multe variante. Poate să vadă care sunt telefoanele disponibile pentru a fi cumpărate și poate continua, dacă își dă acordul, să cumpere telefonul dorit. Pe parcursul procesului de cumpărare, clientul își poate alege pană la un număr maxim de extra 5 ani de garanție, fiecare an în plus fiind taxat cu câte 100 de dolari în plus față de prețul de bază al telefonului.

Cu toate acestea, chiar dacă prețul telefonului este unul ridicat, clientul poate opta pentru plata prețului întreg sau plata în rate egale. Dacă alege plata în rate, acesta poate opta pentru plata în 3, 6, 9 sau maxim 12 rate egale. Când procesul a fost finalizat, iar clientul plasează comanda, în fișierul BoughtPhones.txt se actualizează lista și se adaugă telefonul cumpărat.

Printre alte lucruri pe care le poate face utilizatorul în aplicație sunt vizualizarea informațiilor personale, schimbarea parolei sau chiar filtrarea telefoanelor după anumite criterii pentru a-i fi mai ușor de selectat telefonul dorit.

Ca administrator, acesta poate adăuga un telefon nou în listă sau poate vedea anumite statistici precum venitul total în urma tuturor comenzilor plasate. Meniurile sunt oarecum aceleași, dar sunt personalizate în funcție de permisiunile de acces pe care le are fiecare rol în parte.

# Concluzie

Cu toate acestea, aplicația are rolul de a crea un mediu prin care un client să își poate achiziționa un telefon mobil din lista de produse pusă la dispoziție de aplicație. Atât utilizatorul cât și administratorul au la dispoziție un meniu interactiv prin care pot efectua mai multe comenzi, dar fiecare fiind personalizat în funcție de rol. Plasarea comenzilor este una ușor de realizat, iar toate datele sunt salvate în fișiere, fiind o metodă flexibilă de gestionat.

# Bibliografie

Munteanu, H. S. *GitHub*. Preluat de pe https://github.com/horiasteff/Proiect-PPOO

Ciurea, C. *GitHub*. Preluat de pe https://github.com/cristianciurea/PPOO2022

StackOverflow. Preluat de pe https://stackoverflow.com/questions/3481828/how-do-i-split-a-string-in-java

StackOverflow. Preluat de pe https://stackoverflow.com/questions/822150/modify-a-txt-file-in-java

Baeldung. Preluat de pe https://www.baeldung.com/java-new-custom-exception