UNIVERSITATEA “STEFAN CEL MARE”, SUCEAVA

FACULTATEA DE INGINERIE ELECTRICA SI STIINTA CALCULATOARELOR

SPECIALIZAREA CALCULATOARE

PROIECT

DISCIPLINA POO

**Tema:** Alegerea unui model de laptop  
 **Autor:** Alexandroaie Valentin

Tema Proiect

TEMA SI MOTIVATIA ALEGERII

Tema pe care am ales-o este: “Alegerea unui model de laptop”, are ca rol căutarea sau adăugarea unui laptop în funcţie de dorinţa utilizatorului.

Motivul care m-a împins spre alegerea acestei teme este pasiunea pentru construirea diferitelor configuraţii de laptop-uri sau calculatoare în funcţie de dorinţa fiecărui om. Chiar dacă alegerea poate părea uşoară, compatibilitatea dintre piese poate duce la contrariul acestei păreri. Alegerea unor piese are nevoie de timp, timp pe care trebuie să îl folosim pentru a ne informa corect.

**CUPRINS**

**TEMA ȘI MOTIVATIA ALEGERII**

1. **CAPITOLUL I – ELEMENTE TEORETICE**  
    1.1. Descrierea problemei  
    1.2. Abordarea teoretică a problemei  
    1.3. Elemente specifice POO  
    1.4. Alte capitole specifice
2. **CAPITOLUL II – IMPLEMENTARE**  
    2.1. Tehnologii folosite  
    2.2. Diagrama de clase  
    2.3. Implementarea funcționalităților specifice
3. **CAPITOLUL III – ANALIZA SOLUȚIEI IMPLEMENTATE**  
    3.1. Formatul datelor de I/O  
    3.2. Studii de caz  
    3.3. Performanțele obținute

**CAPITOLUL I – ELEMENTE TEORETICE**

**1.1. DESCRIEREA PROBLEMEI**

Proiectul constă în dezvoltarea unei aplicații C++ pentru alegerea unui model de laptop, care să permită utilizatorului să parcurgă un catalog de laptopuri, să adauge noi modele, să efectueze căutări după diverse criterii (nume, memorie RAM, rezoluție display, placă grafică, preț, rating) și să vizualizeze detalii precum comentarii și ratinguri reprezentate prin stele. Această soluție va ajuta la luarea unei decizii informate în alegerea laptopului.

**1.2. ABORDAREA TEORETICA A PROBLEMEI**

Problema se abordează prin utilizarea paradigmei programării orientate pe obiect, creând clase care modelează entitățile relevante:

* **Laptop** – cu atribute precum nume, model, specificații tehnice, preț și rating.
* **Review** – pentru stocarea comentariilor și ratingurilor (sub formă de stele).
* **Catalog** – pentru gestionarea și manipularea unei colecții de laptopuri.

Se vor utiliza tehnici de citire/scriere din fișiere pentru stocarea datelor, funcționalități de filtrare și sortare și, opțional, o interfață text (meniu) pentru interacțiune.

**1.3. ELEMENTE SPECIFICE POO**

În proiect vor fi aplicate elemente fundamentale ale programării orientate pe obiect, cum ar fi

* **Încapsularea:** Atributele claselor vor fi protejate prin metode de acces (getteri/setteri) și funcții dedicate.
* **Moștenirea:** Posibilă extindere a claselor pentru a adăuga funcționalități specifice (de exemplu, clase derivate pentru categorii particulare de laptopuri).
* **Polimorfismul:** Implementarea unor metode virtuale pentru a personaliza afișarea și procesarea datelor.

**1.4. ALTE CAPITOLE SPECIFICE**

* Modul de gestionare a fișierelor (citire și scriere) pentru persistenta datelor.
* Interfața text.

**CAPITOLUL II – IMPLEMENTARE**

**2.1. TEHNOLOGII FOLOSITE**

* **Limbajul de programare:** C++
* **Mediu de dezvoltare:** Code::Blocks
* **Biblioteci standard:** iostream, fstream, vector, string, algorithm
* **Tehnici suplimentare:** Programare orientată pe obiecte , manipularea fișierelor text

**2.2. DIAGRAMA DE CLASE**

Diagrama UML de mai jos prezintă clasele principale și relațiile dintre ele:

**2.3. IMPLEMENTAREA FUNCȚIONALITĂȚILOR SPECIFICE**

Funcționalități propuse:

* **Căutare după criterii:** Permite filtrarea laptopurilor după nume, memorie RAM, display, placă grafică, preț și rating.
* **Stocarea datelor:** Introducerea și salvarea datelor despre laptopuri într-un fișier extern.
* **Comentarii și rating:** Adăugarea de comentarii și rating sub formă de stele pentru fiecare laptop.
* **Meniu interactiv:** Pentru navigarea între opțiuni precum adăugare laptop, afișare catalog, căutare și sortare după preț.

**CAPITOLUL III – ANALIZA SOLUȚIEI IMPLEMENTATE**

**3.1. FORMATUL DATELOR DE I/O**

Intrare:

* Datele de intrare vor fi introduse manual prin meniul interactiv sau citite dintr-un fișier text care conține informații despre laptopuri.
* Formatul datelor include câmpuri precum: nume, model, procesor, memorie RAM, display, placă grafică, preț, rating și comentarii.

Ieșire:

* Datele vor fi afișate în consolă, prezentând informațiile complete despre fiecare laptop, inclusiv comentariile și ratingurile.

**3.2. STUDII DE CAZ**

Cazuri reprezentative ce vor fi prezentate:

* Adăugarea unui laptop nou și verificarea corectitudinii datelor.
* Căutarea unui laptop după criterii multiple
* Sortarea catalogului după preț și analiza performanței algoritmului de sortare.

**3.3. PERFORMANȚE OBTINUTE**

Se va analiza eficiența operațiunilor de căutare și sortare asupra unui număr mare de înregistrări, evaluând timpii de execuție și complexitatea algoritmilor utilizați.