

## EXEMPLE PROIECTE DE SEMESTRU

=====

### 1. Jumbo

Design-ul si implementarea unei aplicatii pentru gestionarea unui magazin de jucarii. Aplicatia va avea doua tipuri de utilizatori (cumparator si administrator).

Cumparatorul va putea realiza urmatoarele operatii:

- Inregistrarea in aplicatie.
- Vizualizarea produselor.
- Sortare produse in functie de pret.
- Adaugarea produselor in cos.
- Plasarea unei comenzi.
- verificare status comanda.

Administratorul aplicatiei va putea realiza urmatoarele operatii:

- CRUD useri.
- CRUD produse .
- CRUD comenzi.
- Adaugarea de promotii la produse.

### 2. UNTOLD

Design-ul si implementarea unei aplicatii pentru gestionarea vanzarilor de bilete ale unui festival. Aplicatia va avea doua tipuri de utilizatori (cumparator si administrator).

Cumparatorul va putea realiza urmatoarele operatii:

- Inregistrarea in aplicatie.
- Vizualizarea pachetelor de bilete ( bilete pe o zi, bilete pe 2 zile, bilete pe intreg festivalul) .
- Plasarea unei comenzi.
- Verificare status comanda.

Administratorul aplicatiei va putea realiza urmatoarele operatii:

- CRUD useri.
- CRUD bilete.
- CRUD comenzi.
- Trimitere email de confirmare in momentul cumpararii biletelor.
- Trimitere email catre userii din platforma pentru anuntarea vanzarii biletelor.
- Adaugarea de promotii la bilete.

### 3. Wizz Air

Design-ul si implementarea unei aplicatii pentru gestionarea vanzarilor de bilete de avion. Aplicatia va avea doua tipuri de utilizatori (cumparator si administrator).

Cumparatorul va putea realiza urmatoarele operatii:

- Inregistrarea in aplicatie.
- Selectarea unei rute – afisare date disponibile (selectare ruta dus-intors) .
- Adaugare in cos.
- Plasare comanda (aplicarea unei voucher).

Administratorul aplicatiei va putea realiza urmatoarele operatii:

- CRUD useri.
- CRUD bilete.
- CRUD destinatii.
- Rapoarte format CSV pentru cate bilete s-au vandut in functie de ruta.
- Adaugarea de promotii la bilete.

### 4. eMAG

Design-ul si implementarea unei aplicatii pentru gestionarea unui magazine online. Aplicatia va avea doua tipuri de utilizatori (cumparator si administrator).

Cumparatorul va putea realiza urmatoarele operatii:

- Inregistrarea in aplicatie.
- Vizualizarea produselor.
- Filtrare produse in functie de categorie.
- Adaugarea produselor in cos.
- Posibilitatea stingerii produselor din cos; modificarea cantitatii produselor din cos.
- Plasarea unei comenzi.
- Adaugare review la un anumit produs.
- Verificare status comanda.

Administratorul aplicatiei va putea realiza urmatoarele operatii:

- CRUD useri.
- CRUD produse.
- CRUD comenzi.
- Adaugarea de promotii la produse.
- Trimitere facture prin email.
- Raport pe fiecare luna cate produse s-au vandut, in format CSV.

## **5. SINU**

Design-ul si implementarea unei aplicatii de tip SINU. Aplicatia va avea 3 tipuri de utilizatori (student, professor si administrator).

Studentul va putea realiza urmatoarele operatii:

- Logarea in aplicatie.
- Vizualizarea notelor.

Profesorul va putea realiza urmatoarele operatii:

- Logarea in aplicatie.
- Asignarea de note studentilor la materia corespunzatoare la care acesta a predat.

Administratorul aplicatiei va putea realiza urmatoarele operatii:

- CRUD profesori ( adaugare profesori in system, stergere profesori, editare).
- CRUD studenti.
- Descarcarea notelor in functie de grupa/materie/an, in format CSV.
- Trimitere email la anumiți profesori/studenti.

## **6. Aplicatie pentru vizualizarea rezultate sportive**

Design-ul si implementarea unei aplicatii pentru urmarirea rezultatelor sportive (ex. FlashScore). Aplicatia va avea 2 tipuri de utilizatori (user normal si administrator).

Un user va putea realiza urmatoarele operatii:

- Vizualizarea tuturor categoriilor (tenis, fotbal, etc).
- Vizualizarea meciurilor(fiecare meci avea un status(not started, in progress, finished).
- Adaugarea meciurilor de interes in bookmark ( astfel incat la modificarea rezultatului sa fie notificat – ori direct in aplicatie ori prin email).

Administratorul aplicatiei va putea realiza urmatoarele operatii:

- CRUD categorii.
- CRUD meciuri.
- Trimitere email cu rezultatul final in functie de un meci anume.
- Raport al rezultatelor meciurilor din ziua respectiva in functie de categorie, in format CSV.

## 7. OLX

Design-ul și implementarea unei aplicații de tip vânzare/cumpărare (ex. OLX). Aplicația va avea două tipuri de utilizatori (user normal și administrator).

User-ul va putea realiza următoarele operații:

- Înregistrarea în aplicație.
- Vizualizarea anunțurilor.
- Filtrare anunțuri în funcție de categorie.
- Trimitere mesaj către un alt user pentru o postare curentă.
- Postare anunț.
- Editare anunț.
- Stergere anunț.

Administratorul aplicației va putea realiza următoarele operații:

- CRUD useri.
- CRUD anunțuri.
- CRUD categorii .
- Trimitere email către un anumit user selectat.
- Raport pe fiecare luna pentru câte anunțuri au fost postate, în format CSV.

## 8. Marty

Proiectați și implementați o aplicație utilizată pentru înregistrarea comenzilor, gestionarea plăților și gestionarea activității angajaților dintr-un restaurant. Sistemul permite (i) ospătarilor să înregistreze comenzile în sistem și să gestioneze plățile luând în considerare reducerile în funcție de loialitatea clientului, (ii) angajaților din bucătărie să trimită notificări ospătarilor atunci când mâncarea comandată este gata și (iii) unui manager să efectueze operațiuni CRUD asupra informațiilor angajaților și să verifice activitatea acestora. O comandă are următoarele atribute: numărul mesei, ID-ul clientului, ID-ul chelnerului, lista produselor comandate (băuturi, mâncare, desert etc.), precum și cantitatea acestora. Loialitatea unui client este evaluată în funcție de numărul de comenzi pe care acesta le-a comandat de la restaurant, precum și de costul fiecărei comenzi.

## 9. Agenție de turism

Proiectați o aplicație pentru gestionarea activității într-o agenție de turism. Aplicația este utilizată de angajații agenției și le permite acestora să adauge/modifice/elimine vacanțe, să adauge/modifice informații despre clienți și să rezerve vacanțe pentru clienți. Vacanțele pot fi rezervate în țară și în străinătate. Există trei tipuri de vacanțe care pot fi rezervate: croaziere, excursii și sejururi. În cadrul unui sejur, un client poate alege să meargă în una sau mai multe excursii turistice. Informațiile despre clienți și vacanțe sunt stocate într-o bază de date care se actualizează periodic în funcție de informațiile (disponibilitatea hotelurilor/călătoriilor etc.) furnizate de către operatorii care colaborează cu agenția. Aceste informații sunt furnizate în fișiere XML, fiecare având un format specific fiecărui operator.

## 10. Hotel

Aplicația trebuie să aibă două tipuri de utilizatori (un utilizator obișnuit, reprezentat de recepționar, și un utilizator administrator) care trebuie să furnizeze un nume de utilizator și o parolă pentru a utiliza aplicația.

**Utilizatorul obișnuit** poate efectua următoarele operațiuni:

- Să calculeze tarifele camerelor. Dacă un client a rezervat o cameră de mai multe ori în trecut, aplică automat o reducere pentru fidelitate.
- Rezervă camere pentru clienți.
- Anularea rezervărilor de camere.
- Emiterea de facturi.

**Administratorul** poate efectua următoarele operațiuni:

- Operațiuni CRUD asupra conturilor de utilizator.
- CRUD camere,
- CRUD servicii,
- Emiterea de rapoarte în diferite formate (xml, txt) cu rezervările de camere dintr-un interval de date ales, clienții care au cel mai mare număr de rezervări etc.

## 11. School Master

Proiectați și implementați o aplicație pentru administrația școlii sau a liceului, care poate fi utilizată de profesori, elevi, părinți și de un administrator.

**Profesorii** pot efectua următoarele operațiuni:

- Să-și vizualizeze orarul.
- Să vizualizeze situația școlară a unui elev.
- Să atribuie note elevilor.
- Finalizați situația elevilor la disciplina pe care o predau.

**Elevii** pot efectua următoarele operațiuni:

- Să-și vizualizeze orarul.
- Vizualizați situația lor școlară.

**Părinții** pot efectua următoarele operațiuni:

- Să vizualizeze orarul copiilor lor.
- Să vizualizeze situația școlară a copiilor lor.

**Administratorul** poate efectua următoarele operațiuni:

- Operațiuni CRUD privind elevii, profesorii și clasele.
- Atribuirea elevilor la clase.
- Atribuirea profesorilor la clase.
- Închiderea situației unui elev la sfârșitul unui semestru.

Sistemul trimite automat un e-mail unui părinte atunci când copilul său primește o notă și când situația acestuia este închisă.

## **12. Car Renting Application**

Proiectați și implementați o aplicație pentru închirierea de autoturisme care poate fi utilizată de clienți și de angajații firmei de închirieri. Aplicația va permite unui client să caute și să vizualizeze mașinile disponibile (informații detaliate despre mașini și imagini) pentru intervalul de timp selectat și din orașul sursă specificat. Pentru a închiria o mașină, clientul trebuie să își completeze datele personale. După ce clientul a închiriat o mașină, i se trimite un e-mail cu detaliile închirierii.

Atunci când un client care a închiriat o mașină vine la compania de închirieri pentru a ridica mașina, un angajat creează un contract care este apoi tipărit și semnat de către client. Atunci când o mașină este adusă înapoi de către un client, angajatul marchează mașina ca fiind returnată.

Aplicația trebuie să aibă, de asemenea, o componentă de administrator pentru operațiuni CRUD asupra mașinilor, clienților și angajaților.

## **13. Building management**

### **14. Car Shop**

### **15. Fitness**

### **16. Flower Shop**

### **17. Glovo – food delivery**

### **18. World Class**

### **19. Cărturești**

### **20. Cinema City**

### **21. Dr. Max**

### **22. Maxi-Pet**