**Tema restanta / marire (septembrie 2018)**

**Invatamant la distanta**

Cei care au situatia neincheiata sau doresc marirea notei de laborator la Medii Vizuale de Programare vor prezenta in perioada de restante din septembrie rezolvarea pentru temele de mai jos.

Aplicatiile trebuie sa fie implementata in limbajul C#, WPF Application, platforma .NET Framework 4.5, Visual Studio 2012 sau mai nou; serverul de baze de date folosit va fi SQL Server 2012 sau mai nou.

**Aplicatia 1:** Sa se realizeze o aplicatie pentru gestiunea unui hotel. Aplicatia va avea 4 tipuri de utilizatori: utilizator neautentificat, administrator, client si angajat. Utilizatorii isi pot crea cont din aplicatie.

**Administratorul** este cel care poate adauga, modifica si sterge camere, dotari pentru tipurile de camere, servicii suplimentare pentru hotel, preturi pentru tipurile de camere si pentru serviciile suplimentare, oferte.

**Utilizatorul neautentificat** poate vizualiza tipurile de camere impreuna cu informatiile aferente (dotari, imagini, preturi, numar de camere disponibile). Acesta poate face si o cautare pentru o anumita perioada si va vizualiza pretul pentru toata perioada aleasa pentru toate tipurile de camere disponibile in perioada respectiva.

**Clientul**, pe langa functionalitatea utilizatorului neautentificat, mai poate face si rezervare. In rezervare se va specifica tipul de camera (tipurile – in cazul in care rezerva mai multe camere de tipuri diferite), numarul de persoane, perioada in care doreste sa se cazeze, daca alege vreo oferta si va vizualiza si costul total al rezervarii. Clientul nu va putea rezerva o camera decat daca aceasta este disponibila in perioada aleasa. Clientul va putea deasemenea sa anuleze o rezervare activa.

Dupa trecerea datei rezervarii, aceasta nu va mai putea fi modificata de catre nici un utilizator.

Clientul va putea adauga la o rezervare si anumite servicii suplimentare, iar pretul se va calcula si impreuna cu preturile serviciilor.

**Angajatul hotelului**, pe langa functionalitatile de utilizator neautentificat, va mai putea vizualiza rezervarile si va putea modifica starea unei rezervari active (prin starea unei rezervari se va intelege: activ, anulat, achitat)

Se vor mai avea in vedere urmatoarele lucruri:

1. Preturile se vor modifica in timp, asa ca, daca rezervarea prinde la mijloc o schimbare de pret, se va calcula cu doua preturi. Odata efectuata rezervarea, pretul calculat va ramane nemodificat.
2. Nu se va face stergerea fizica a datelor din baza de date, ci stergere logica (veti pune in tabelele din baza de date un camp „Activ” care va avea valorile True sau False). Se vor lua in considerare doar datele din tabele, pentru care acel camp are valoarea True.

**Barem de notare:**

1. Adaugare, modificare si stergere conturi de utilizatori (**0.5p**)
2. Adaugare, modificare si stergere tipuri de camere (pentru fiecare tip de camera se va sti numarul de camere, se vor adauga imagini, se vor asocia dotari si se vor stabili preturi) (**1p**)
3. Adaugare, modificare si stergere dotari (**0.5p**)
4. Adaugare, modificare si stergere servicii suplimentare (serviciile vor avea preturi variabile in timp) (**0.5p**)
5. Adaugare, modificare si stergere oferte (oferta va avea o denumire, un numar de zile si o perioada de timp in care va fi valida. **Exemplu**: ‚Litoralul pentru toti” –> 5 nopti cu 400 de lei, in intervalul 1.09.2016 – 30.09.2014) (**0.5p**)
6. Autentificarea corecta a utilizatorilor (**0.5p**)
7. Rezervarea unei camere (impreuna cu efectele pe care le are aceasta: reducerea numarului de camere disponibile intr-o anumita perioada)
   1. Stabilirea numarului si a tipului de camere rezervate, a numarului de persoane si a datelor rezervarii (**0.25p**)
   2. Adaugarea de servicii suplimentare la rezervare (**0.25p**)
   3. Calcularea corecta a costului rezervarii (**0.5p**)
   4. Utilizarea corecta a unei oferte pentru rezervare (**0.5p**)
   5. Posibilitatea de efectuare a rezervarii doar pentru tipuri de camere si perioade disponibile (**0.5p**)
8. Anularea unei rezervari (impreuna cu efectele pe care le are aceasta) (**0.5p**)
9. Achitarea unei rezervari(impreuna cu efectele pe care le are aceasta) (**0.5p**)
10. Vizualizarea tipurilor de camere ale hotelului, impreuna cu toate informatiile aferente (**0.5p**)
11. Cautarea de camere disponibile intr-un anumit interval de timp si afisarea corecta a pretului pentru numarul de zile ales. (**1p**)
12. Oficiu (**2p**)

**Aplicatia 2:** Sa se scrie o aplicatie care sa permita gestiunea unui cabinet stomatologic. Se cunosc informatii despre medici, pacienti, programari, interventii (ex: plomba, extractie, anestezie, obturatie canal, etc...), preturi si utilizatori. Vor fi doua tipuri de utilizatori: medic si administrator, cu atributii diferite.

Acestia se vor autentifica pe aplicatie cu ajutorul unui nume de utilizator si a unei parole. **Medicii** vor putea sa faca programari pentru pacienti, vor putea introduce, modifica si sterge pacientii din sistem.

**Administratorul** prelucreaza informatile despre medici, preturi si interventii, precum si pentru conturile de utilizatori.

Toate stergerile din sistem se vor opera doar la nivel logic, nu si fizic. O stergere logica se refera la faptul ca inregistrarea ramane in sistem, dar este marcata printr-un camp specificat, ca fiind stearsa.

Aplicatia trebuie sa permita:

* 1. Adaugare, modificare si stergere conturi de utilizatori (0.5p)
  2. Autentificarea corecta a utilizatorilor (0.5p)
  3. Adaugare, modificare si stergere medici (0.5p)
  4. Adaugare, modificare si stergere pacienti (pacientii vor fi asociati permanent unui medic, in caz contrar nu se obtine punctajul) (1p)
  5. Adaugare, modificare si stergere interventii (atunci cand se adauga o interventie se face si asocierea cu un pret; pretul are o data de inceput si una de sfarsit; se va pastra istoricul preturilor in timp) (1p)
  6. Adaugare, modificare si stergere programari (orice programare trebuie asociata unui pacient si unui medic, in caz contrar nu se obtine punctaj; programarea se face la o anumita data si ora) (1p)
  7. Folosire data binding (0.5p)
  8. Efectuarea urmatoarelor operatiuni:
     1. sa se afiseze lista cu preturile interventiilor stomatologice ale acelui cabinet (0.5p)
     2. sa se afiseze gradul de ocupare al unui medic intr-o anumita perioada(ce procent din timpul de lucru este ocupat cu programari – se considera ca medicul lucreaza 8 ore pe zi, 5 zile pe saptamana) (0.5p)
     3. sa se sorteze pacientii unui medic in functie de numarul de interventii (se va afisa o lista de forma: nume pacient – numar de interventii) (0.5p)
     4. sa se afiseze castigul medicului pe fiecare luna (0.5p)
     5. sa se afiseze pentru o anumita interventie, preturile mai vechi decat o anumita data (selectata dintr-un calendar) (0.5p)
     6. sa se afiseze suma incasata de medic de la fiecare pacient al sau (0.5p)
     7. sa se afiseze toate interventiile unui pacient, impreuna cu data in care au avut loc (0.5p)
     8. sa se afiseze programarile din ziua curenta ale medicului (0.5p)

**Nota**: Un medic nu are acces la nici o informatie legata de alti medici (nu poate vedea pacientii sau programarile altor medici).

**Obs:** Punctajul pentru fiecare cerinta se va acorda integral doar in cazul in care cerinta a fost executata corect si complet. Aplicatiile care nu au validari pentru date vor fi depunctate (-1p). Aplicatiile care nu au baza de date in FN3 vor fi depunctate (-1p). Aplicatiile care permit SQL Injection vor fi depunctate (-5p). Aplicatiile care nu sunt de tipul WPF Application nu vor fi luate in considerare. Aplicatiile care nu folosesc “Binding” vor fi depunctate (-2p). Aplicatiile care nu sunt structurate folosind MVVM vor fi depunctate (-2p).

**Spor la lucru!!**