

## COMENZI RESTAURANT – 20 Ianuarie 2022

Se doreste o aplicatie cu o interfata grafica intuitiva, care sa permita plasarea comenzilor in cadrul unui restaurant.

### Cerinte functionale - 7p:

- 1) **1p** La pornirea aplicatiei se va deschide **o fereastră pt. angajatii restaurantului si cate o fereastră pt. fiecare masa citita din fisierul/tabelul Tables**. Fereastră destinata angajatilor va afisa numele “Staff”, si fiecare fereastră asociata unei mese va afisa id-ul acesteia (ex: “Table 3”).

- Se va defini clasa:

**Table:** { **id:** int }

- 2) **2p** Fiecare dintre ferestrele asociate meselor va afisa **meniul restaurantului, grupat pe categorii**. Meniul se va citi din fisierul/tabelul **Menu**.

- Se va defini clasa:

**MenuItem:** { **id:** int, **category:** String, **item:** String, **price:** float, **currency:** String }

- Ex. de intrari din fisierul/tabelul Menu:

1, Antreuri, Bruschete cu rosii, 15, RON  
2, Antreuri, Salata Caprese, 20, RON  
3, Fel Principal, Paste cu sos pesto, 25, RON  
4, Fel Principal, Vinete parmigiana, 25, RON

- Meniul va fi afisat sub forma de tabele, cate unul pentru fiecare Categorie. De ex:

*Antreuri*

|                    |        |
|--------------------|--------|
| Bruschete cu rosii | 15 RON |
| Salata Caprese     | 20 RON |

*Fel Principal*

|                    |        |
|--------------------|--------|
| Paste cu sos pesto | 25 RON |
| Vinete parmigiana  | 25 RON |

- 3) **2p** In ferestrele asociate meselor, clientii pot **selecta mai multe intrari din meniu** (prin multiple selection sau checkbox-uri etc.), iar apoi pot **plasa comanda** aferenta produselor selectate prin actionarea unui buton “Place order”.

Comenzile plasate se vor salva in fisierele/tabelele Orders si OrderItems, avand data calculata automat si statusul PLACED.

*Observatie: Pt. simplitate vom considera ca o comanda poate continue o singura bucata dintr-un anumit produs.*

- Se vor defini clasele:

**Order:** { **id:** int, **table:** int/Table, **menuitems:** List<Integer>/List<MenuItem>, **date:** LocalDateTime, **status:** OrderStatus }, unde OrderStatus este un enum {PLACED, PREPARING, SERVED}

- Ex. de continut fisiere/tabele Orders si OrderItems:

*Order:*

*id=1, tableId=3, date=2022-01-20T12:00, status=PLACED*

*OrderItems:*

*orderId=1, menuItemId=1*

*orderId=1, menuItemId=2*

- 4) **2p** In momentul in care o comanda este plasata aceasta va aparea **instantaneu** (fara a fi nevoie de vreo actiune din partea utilizatorului) **in fereastră “Staff”**, in cadrul unui tabel Placed Orders. Intrarile din acest tabel vor fi ordonate crescator dupa data la care s-a plasat comanda. Se vor afisa: id-ul mesei, data, numele produselor comandate (nu id-urile!).

### Cerinte non-functionale - 2p:

- Validarea datelor de intrare  
Exemplu: Nu se poate plasa o comanda fara a selecta niciun produs
- Procesarea va avea loc numai la nivel de service sau de controller; interactiunea cu sursa de date se va face numai prin intermediul repository-ului (puteti folosi baze de date sau fisiere text)
- Interactiunea cu utilizatorul va avea loc numai in UI (GUI)
- Se cere eliminarea codului care nu este folosit, precum si a functionalitatilor care nu s-au cerut (daca ati lucrat cu ceva template de la lab)
- Clasele, atributele și metodele lor vor avea exact numele cerute in problema sau nume sugestiv daca nu s-a specificat explicit numele lor

### IMPORTANT

- Se puncteaza doar cerintele functionale care ruleaza
- Orice cod care nu poate fi explicat, atrage dupa sine nepunctarea cerintei/cerintelor din care face parte
- Nu aveti voie sa comunicati in timpul examenului, in nicio modalitate posibila de a face acest lucru (chat, mail, ...etc.)

### Cerinte extra – pt. bonus: 1p in plus la nota finala:

Pt. 1p in plus la nota finala se poate rezolva oricare dintre urmatoarele cerinte (nu ambele):

- 1) Adaugarea posibilitatii de a selecta cantitatea dorita pt. fiecare produs atunci cand se plaseaza o comanda.
  - Se va adauga clasa **OrderItem** { **order**: int/Order, **menuItem**: int/MenulItem, **quantity**: int }
  - Field-ul **menuItems** din clasa **Order** va continue o lista de id-uri de OrderItem sau o lista de OrderItem
  - Pentru fiecare intrare din fisierul/tabelul OrderItems se va adauga si cantitatea
  - Pe fiecare rand din tabelele in care este afisat meniul se va adauga o noua coloana ce va continue un input field "Quantity". Acesta va fi disabled pt. produsele care nu au fost selectate.
- 2) Fereastra "Staff" va permite schimbarea statusului comenzilor in felul urmator:
  - O comanda PLACED poate fi marcata ca PREPARING DELIVERED (prin actionarea unui buton). In acest caz comanda se va muta din tabelul Placed Orders intr-un nou tabel Preparing Orders (ordonat tot dupa data ascending). Fereastra corespunzatoare mesei asociata comenzii va afisa instantaneu o notificare "Your order is being prepared"
  - O comanda PREPARING (din tabelul Preparing Orders) poate fi marcata ca DELIVERED (prin actionarea unui buton). In acest caz comanda va disparea din tabelul Preparing Orders. Fereastra corespunzatoare mesei asociata comenzii va afisa instantaneu o notificare "Your order has been delivered to your table"
  - Toate aceste modificari de status vor fi salvate corespunzator in fisierul/tabelul Orders

Timp de lucru: 1h 30min.

1p din oficiu