

Vîlculescu Mihai-Bogdan  
Martac Dana-Maria  
Jurcăneanu Ionuț-Adrian  
Drăguțescu Mihai-Valentin  
Borș Andrei

Grupa 232

**Specificații tehnice pentru  
Restaurant  
Admin**

## Cuprins

<b>1. Introducere</b>	<b>3</b>
1.1. Scurtă descriere	3
1.2. Obiective	3
<b>2. Cerințe software</b>	<b>4</b>
2.1. Sistemul de operare	4
2.2. Conexiune la internet	4
2.3. Evoluția aplicației	4
<b>3. Specificații tehnice</b>	<b>5</b>
3.1. Interfața de Autentificare	5
3.2. Meniu principal (Administrator)	6
3.3. Vezi Angajați	7
3.4. Adaugă Nou Angajat	8
3.5. Administrare sala	9
3.6. Reîncarcă planificare	11
3.7. Vezi Meniu	12

# 1. Introducere

## 1.1. Scurtă descriere

*Restaurant-Admin* este o aplicație mobil care funcționează pe sistemul de operare Android. Scopul său este acela de a simplifica lucrul într-un restaurant.

În orice restaurant apare destul de des problema menținerii ordinii comenzilor și a meselor. Din această cauză, în astfel de situații se creează haos. Pentru a preveni aceste momente nefavorabile afacerii și relațiilor inter-umane, aplicația *Restaurant-Admin* oferă facilități pentru administrarea eficientă a afacerilor interne ale unui restaurant.

În acest document, sunt detaliate obiectivele aplicației și facilitățile puse la dispoziție de aceasta. De asemenea, vor fi prezentate și specificațiile tehnice ale produsului software.

## 1.2 Obiective

Aplicația are ca principal obiectiv facilitarea lucrului într-un restaurant folosind doar un simplu smartphone. Pentru a putea îndeplini obiectivul acesta, aplicația permite:

- ★ Autentificarea angajaților restaurantului;
- ★ Diferențierea între mai multe tipuri de angajați: *Administrator*, *Ospătar*, *Bucătar*;
- ★ Administratorul poate gestiona angajații și le poate crea conturile;
- ★ Tot administratorul poate administra un layout al așezării meselor în sală pentru restaurantul său și mai poate gestiona și meniul pus la dispoziție clienților;
- ★ Ospătarul poate vedea așezarea meselor în sală și poate gestiona comenzile în funcție de masă.

## 2. Cerințe software

### 2.1. Sistemul de operare

Aplicația a fost concepută pentru a rula pe smartphone-urile cu sistem de operare *Android*. Versiunea de *Android* minimă necesară funcționării optime este de *Android 15*.

### 2.2 Conexiune la internet

*Restaurant-Admin* are la bază o continuă comunicare cu un server extern. Totuși, aplicația are implementată și o bază de date locală unde sunt copiate datele de pe serverul extern pentru a permite funcționarea software-ului și *offline*.

Pentru autentificarea utilizatorilor este necesară însă existența unei conexiuni la internet. Din această cauză, pentru o *funcționare optimă* și pentru a minimiza eventualele probleme, *recomandăm* menținerea unei conexiuni la internet pe parcursul utilizării aplicației.

### 2.3 Evoluția aplicației

Ca orice produs software, aplicația *Restaurant-Admin* va fi dezvoltată pe viitor pentru a putea simplifica și mai mult lucrul într-un restaurant. Ne dorim ca mai departe să implementăm mai multe feature-uri, precum:

- ★ Bucătarul să poată primi o listă cu comenzile în desfășurare și să aleagă prepararea uneia dintre ele;
- ★ Crearea unui nou tip de utilizator, *clientul*;
- ★ Clientul va putea să facă rezervări prin intermediul aplicației;

## 3. Specificații tehnice

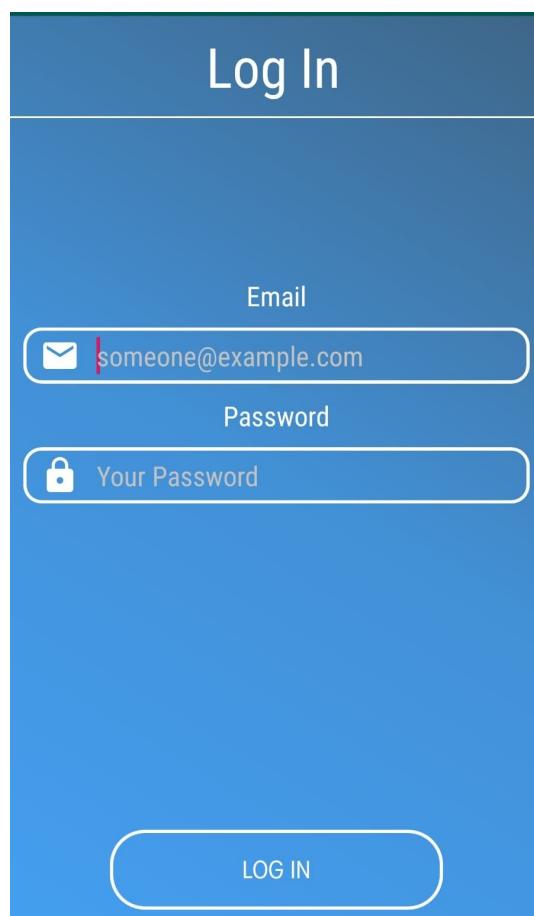
### 3.1. Interfața de Autentificare

Fereastra de *Autentificare* este prima fereastră cu care intra în contact utilizatorul la prima sa logare sau în cazul în care acesta s-a deconectat la ultima utilizare a aplicației. În această situație este necesară completarea câmpurilor *Email* și *Password* (parola de minim 6 caractere) care vor fi verificate în baza de date la apăsarea butonului “LOG IN”. Pentru lucrul cu baze de date am folosit platforma “*Firebase*”.

În cazul în care utilizatorul a rămas conectat de la ultima folosire a aplicației, acesta va fi redirecționat automat către fereastra cu meniul principal. În funcție de tipul utilizatorului identificat (*Administrator*, *Bucătar* și *Ospătar*) va apărea un meniu diferit.

Pentru a putea transmite datele obiectului de tip user mai departe către activitatea de maniu principal, am folosit procedeul de *serializare*.

Administratorului i se va crea contul înainte de prima utilizare a aplicației, iar conturile celorlalți angajați se pot crea de către acesta în aplicație după cum vom putea vedea mai jos.



The image shows a mobile application login screen. At the top, there's a dark blue header with the text "Log In" in white. Below the header, the background is a lighter blue. There are two input fields: the first is labeled "Email" and contains the text "someone@example.com" with a red cursor; the second is labeled "Password" and contains the text "Your Password". Both fields have a white border and a small icon on the left (an envelope for email and a lock for password). At the bottom, there is a white rounded rectangular button with the text "LOG IN" in blue capital letters.

### 3.2. Meniu principal (Administrator)

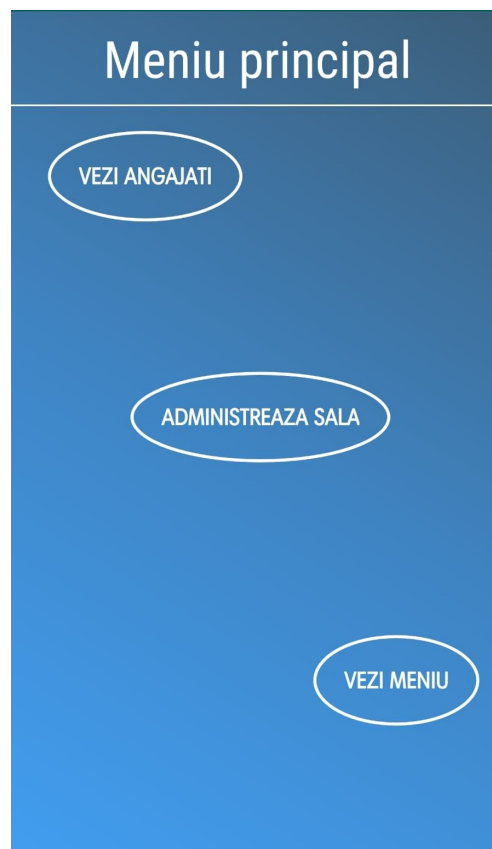
Interfața ferestrei de meniu principal este generată în funcție de tipul utilizatorului conectat (*Administrator*, *Bucătar* sau *Ospătar*). Pentru utilizatorul de tip *Administrator*, interfața arată ca în imaginea din dreapta și oferă următoarele opțiuni:

- ★ Vezi Angajați;
- ★ Administrează Sala;
- ★ Vezi Meniu.

Butonul *Vezi Angajați* permite accesul la lista angajaților din restaurant și modificarea datelor acestora.

Butonul *Administrează Sala* permite accesul la o activitate orientată orizontal (landscape), care permite crearea zone de lucru.

Butonul *Vezi Meniu* permite accesul la vizualizarea și modificarea meniului restaurantului.



### 3.3 Vezi Angajați

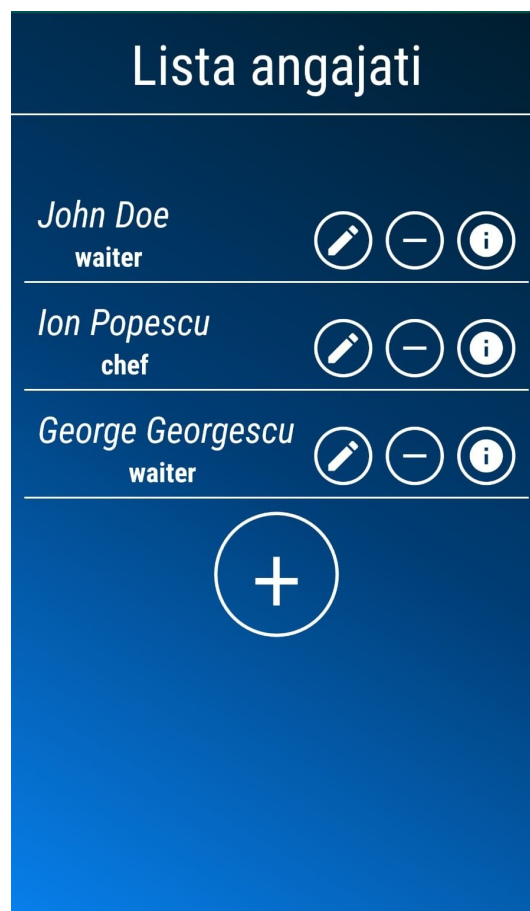
Această fereastră are rolul de a afișa toți angajații adăugați de către administrator până în prezent. Pentru aceasta s-a folosit un *RecyclerView* în care au fost introduse, pentru fiecare element (*Card*), numele angajatului, job-ul și 3 butoane cu următoarele funcționalități:

- ★ modificarea datelor angajatului;
- ★ eliminarea angajatului din baza de date;
- ★ detalii despre angajat;

Toate aceste modificări sunt salvate în baza de date.

Butonul Plus permite adăugarea unui nou angajat prin deschiderea ferestrei *Adaugă Nou Angajat*.

Dacă nu exista niciun angajat se va afișa un mesaj corespunzător.



### 3.4 Adaugă Nou Angajat

Fereastra *Adauga Nou Angajat* permite adăugarea de noi angajați în baza de date prin completarea următoarelor câmpuri:

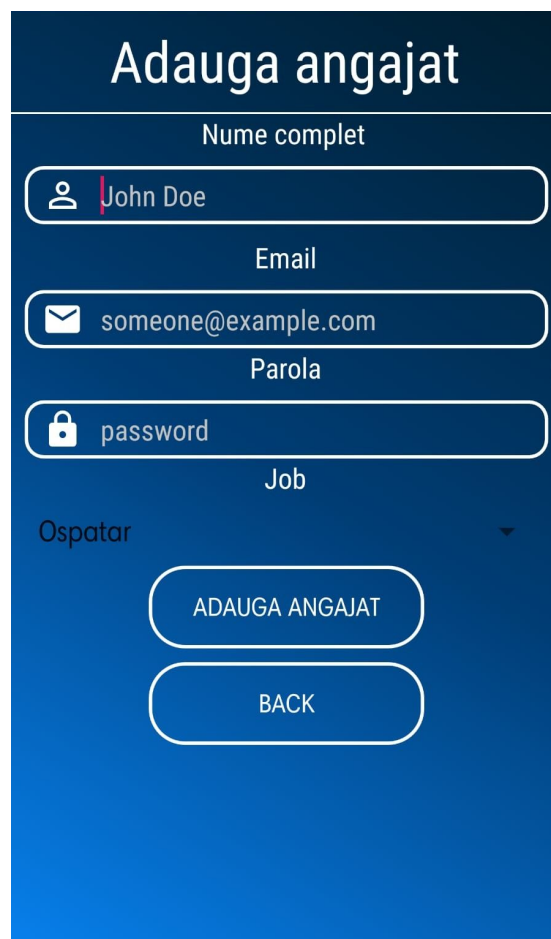
- ★ Nume complet: trebuie respectat formatul "Nume Prenume";
- ★ Email;
- ★ Parola: trebuie să aibă minim 6 caractere și poate fi modificată la prima utilizare a angajatului

După este necesară selectarea jobului angajatului dintre cele doua variante:

- ★ Ospătar;
- ★ Bucătar.

După completarea datelor, dacă a fost respectat formatul specificat, se va adăuga în baza de date noul angajat prin apăsarea butonului *Adauga Angajat*. Mai întâi este creat contul angajatului utilizând email-ul și parola furnizate, iar apoi sunt inserate în tabela *Employees* informațiile acestuia.

Butonul *Back* ne redirecționează către fereastra anterioară *Vezi Angajați*.



Adauga angajat

Nume complet

John Doe

Email

someone@example.com

Parola

password

Job

Ospatar

ADAUGA ANGAJAT

BACK



### 3.5 Administrare sala

Această fereastră permite administratorului să creeze sau să modifice planificarea restaurantului cu ajutorul unor imagini sugestive plasate într-o legenda de elemente în partea stânga.

Prin apăsarea butonul *Începe* sunt calculate dimensiunile matricii ce va constitui sala restaurantului în funcție de rezoluția dispozitivului mobil. Totodată este salvată în baza de date configurația sălii ce urmează a fi actualizată în funcție de preferințele utilizatorului.

Butonul *Șterge* va elimina toate datele referitoare la configurația sălii din baza de date. Butonul *Urmator* ne redirecționează către fereastra *Reîncarcă Planificarea*.

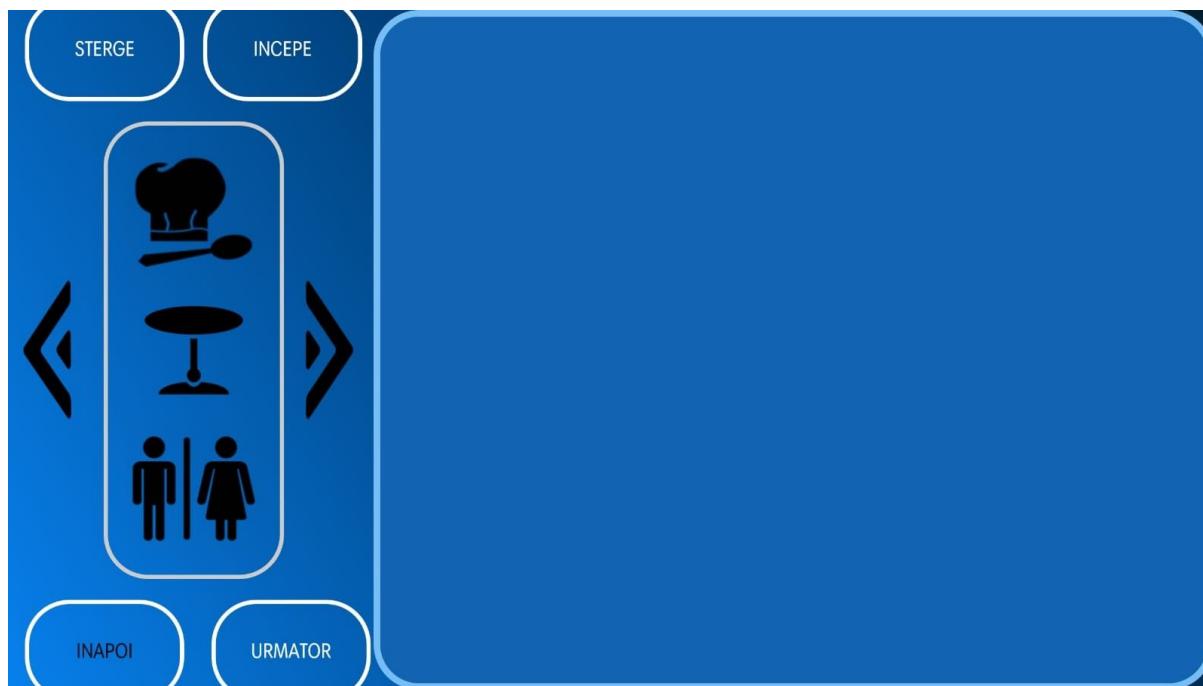


Fig.1 Sala goala

Pentru amplasarea elementelor în sala am folosit o clasa ce implementează metoda *onTouch* din interfața *View.OnTouchListener*, metoda ce detectează începutul evenimentului *drag and drop*. Astfel când se dorește adăugarea unei noi imagini în planificare se detectează acest eveniment și se salvează id-ul imaginii alese.

Atunci când imaginea este lăsată deasupra unei celule se tratează cazul *drop* al metodei *onDrag* din interfața *View.OnDragListener* prin care actualizăm celula cu imaginea respectivă și câmpul *Conținut* din tabela *Celule* ce reprezintă numele imaginii noi adăugate.

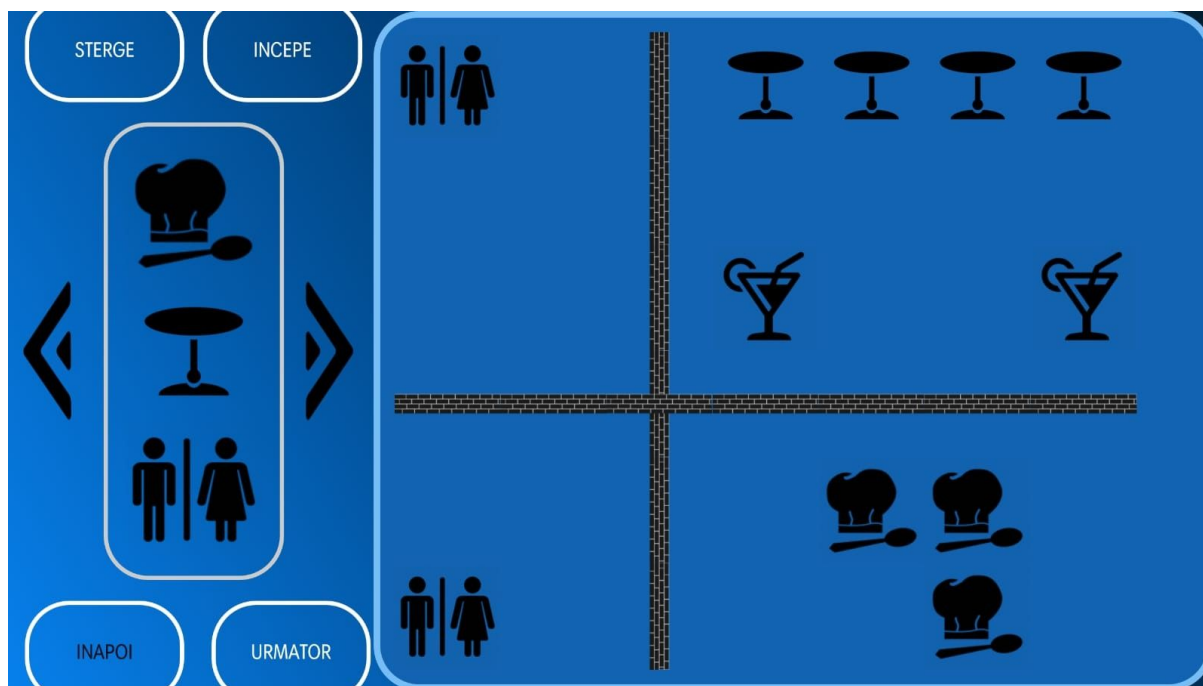


Fig.2 Model de configurație a sălii

### 3.6 Reîncarcă planificarea

La apăsarea butonului *Reîncarca* se va reconstrui ultima planificare a restaurantului prin accesarea fiecărei celule din baza de date și afișarea acestora într-un *RelativeLayout* pe pozițiile corespunzătoare.

La apăsarea pe o celula ce conține o masă, va apărea o fereastră Pop-Up prin care se vor putea efectua acțiuni precum *Adaugă/Anulează comanda* sau *Mută comanda la altă masă*.

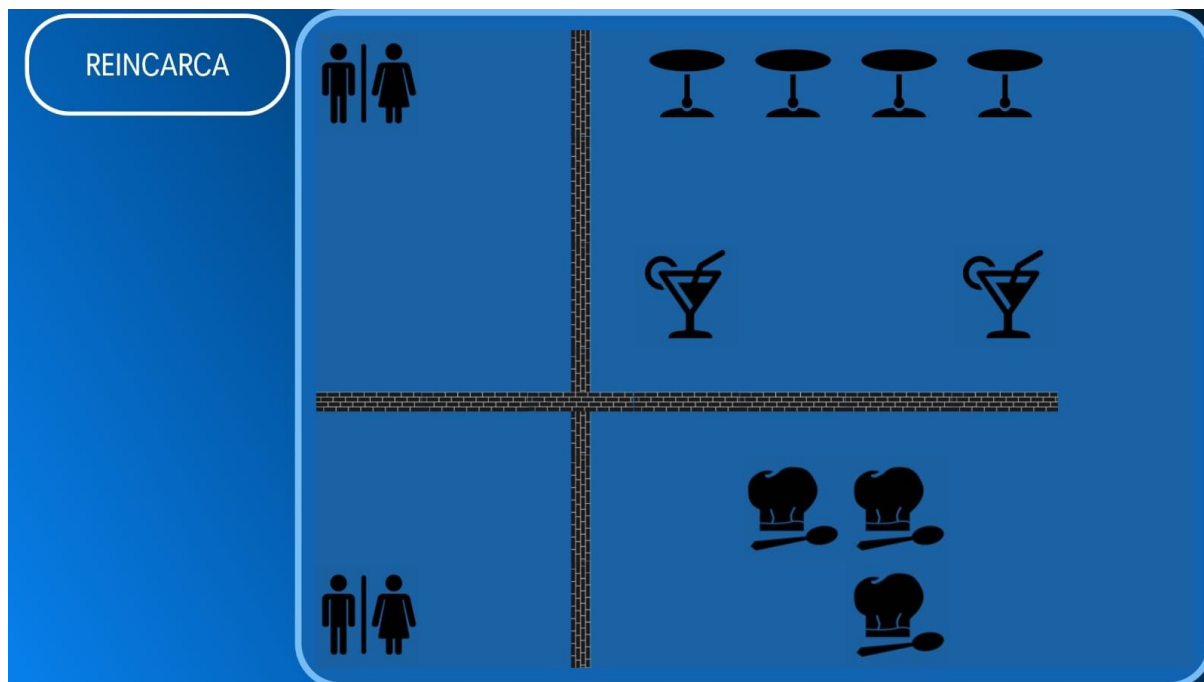


Fig.2 Model de configurație a sălii(REÎNCĂRCARE)

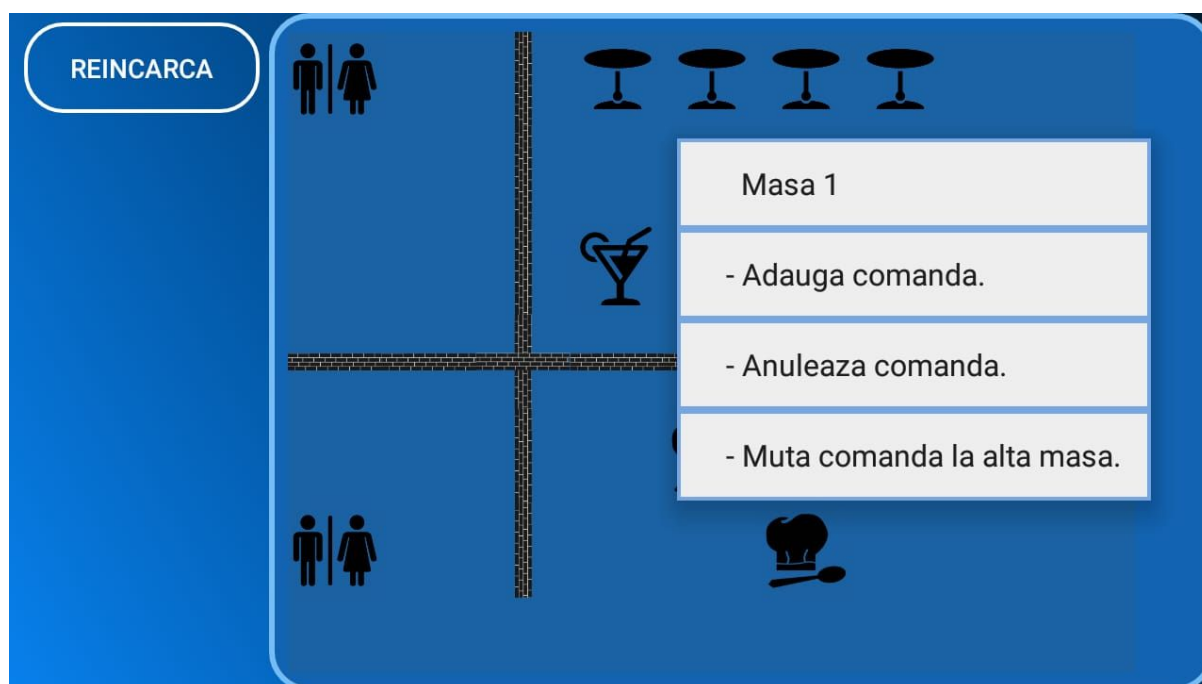


Fig.3 Model de configurație a sălii(REÎNCĂRCARE) + Fereastră Pop-Up

### 3.6 Vezi Meniu

Fereastra *Vezi Meniu* are rolul de a afișa toate preparatele meniului din baza de date împreună cu ingredientele, cantitatea și prețul acestora. Această fereastră funcționează asemănător cu fereastra *Vezi Angajați*.

Butonul Plus permite adăugarea unui nou produs în baza de date

