**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE**

**CATEDRA CALCULATOARE**

**Proiect de Semestru**

**La Disciplina**

**Intorducere în Baze de Date**

**An academic: 2022-2023**

**Nume și grupă:**

**Toderean Razvan Florin Grupa 30222**

1. **Diagrama UML**
2. **Bonusuri**
3. **Lista Proceduri**
4. **Lista Populare**
5. **Lista Tabele**
6. **Manual de utilizare**
7. **Diagrama UML**

**Diagram

Description automatically generated**

1. **Efectuarea bonusurilor**

**Pentru a repartiza automat am transmis in tabela repartizari o cheie straina la cabinet si o cheie straina la CNP-ul fiecarui medic astfel incat repartizarea sa fie automata.**

**Iar pentru calcularea salariului am adunat suma fiecarui serviciu pe care il poate face un medic la salariul initial.**

1. **Lista proceduri**

**drop procedure if exists addServiciu;**

**DELIMITER \\**

**create procedure addServiciu(cnpMedic int(20), serviciuMedical varchar(100),domeniu varchar(100), pret int, durata int, out s int)**

**begin**

**set s := 0;**

**set @flag := null;**

**set @flag := (select idServiciu from servicii where cnp = cnpMedic and serviciu like serviciuMedical);**

**if (@flag is null) then**

**insert into servicii(cnp, serviciu,specialitate, pret, durata) VALUES (cnpMedic, serviciuMedical,domeniu, pret, durata);**

**set s := 1;**

**end if;**

**end; \\**

**DELIMITER ;**

**drop procedure if exists addProgramare;**

**DELIMITER \\**

**create procedure addProgramare(numeMedic varchar(255), numeP varchar(255), prenumeP varchar(255), cnpMedic int(20), serviciuF varchar(100), unit varchar(255),ora time,durata time, zi date, out s int)**

**begin**

**set s:= 0;**

**set @flag := ora;**

**set @orar := null;**

**set @CNP := (select cnp from user1 where nume like numeMedic);**

**set @ziP := dayofweek(zi);**

**case**

**when @ziP = 1 THEN SET @orar := (select id\_orar from Orare\_generice as o where @CNP = cnp and o.zi like 'duminica'**

**and unit LIKE unit and ora >= ora\_inceput and ora <= ora\_sfarsit);**

**when @ziP = 2 THEN SET @orar := (select id\_orar from Orare\_generice as o where @CNP = cnp and o.zi like 'luni'**

**and unit LIKE unit and ora >= ora\_inceput and ora <= ora\_sfarsit);**

**when @ziP = 3 THEN SET @orar := (select id\_orar from Orare\_generice as o where @CNP = cnp and o.zi like 'marti'**

**and unit LIKE unit and ora >= ora\_inceput and ora <= ora\_sfarsit);**

**when @ziP = 4 THEN SET @orar := (select id\_orar from Orare\_generice as o where @CNP = cnp and o.zi like 'miercuri'**

**and unit LIKE unit and ora >= ora\_inceput and ora <= ora\_sfarsit);**

**when @ziP = 5 THEN SET @orar := (select id\_orar from Orare\_generice as o where @CNP = cnp and o.zi like 'joi'**

**and unit LIKE unit and ora >= ora\_inceput and ora <= ora\_sfarsit);**

**when @ziP = 6 THEN SET @orar := (select id\_orar from Orare\_generice as o where @CNP = cnp and o.zi like 'vineri'**

**and unit LIKE unit and ora >= ora\_inceput and ora <= ora\_sfarsit);**

**when @ziP = 7 THEN SET @orar := (select id\_orar from Orare\_generice as o where @CNP = cnp and o.zi like 'sambata'**

**and unit LIKE unit and ora >= ora\_inceput and ora <= ora\_sfarsit);**

**end case;**

**insert into programare(ziua ,ora ,durata ,CNP ,serviciu ,unitate\_medicala ,nume\_pacient ,prenume\_pacient)**

**values (zi, ora, durata, @CNP, serviciuF, unit, numeP, prenumeP);**

**end;\\**

**DELIMITER ;**

**drop procedure if exists calcSalar;**

**DELIMITER \\**

**create procedure calcSalar(CNPmedic int(20),numeMedic varchar(255), out salariu1 int)**

**begin**

**@aux := NULL;**

**@null := (Select cnp from user1 where nume like numeMedic);**

**if (@aux is not null) THEN**

**set @sum := 0;**

**set @sum := (select SUM(pret) from programare as p join servicii\_medicale as s where p.serviciu like s.serviciu**

**and s.CNP\_medic = @aux);**

**end if;**

**SET salariu1 := @sum;**

**end if;**

**end;**

**DELIMITER ;**

1. **Lista Populare**

**insert into Program(luni\_inceput, luni\_sfarsit, marti\_inceput ,marti\_sfarsit ,miercuri\_inceput ,miercuri\_sfarsit ,joi\_inceput ,joi\_sfarsit ,vineri\_inceput ,**

**vineri\_sfarsit ,sambata\_inceput ,sambata\_sfarsit ,duminica\_inceput ,duminica\_sfarsit )**

**values('07:00:00','14:00:00', '07:00:00','14:00:00', '07:00:00','14:00:00', '07:00:00','14:00:00', '07:00:00','14:00:00'**

**, '07:00:00','14:00:00', '07:00:00','14:00:00'), ('07:00:00','14:00:00', '07:00:00','14:00:00', '07:00:00','14:00:00', '07:00:00','14:00:00', '07:00:00','14:00:00'**

**, '07:00:00','14:00:00', '07:00:00','14:00:00'),('07:00:00','14:00:00', '07:00:00','14:00:00', '07:00:00','14:00:00', '07:00:00','14:00:00', '07:00:00','14:00:00'**

**, '07:00:00','14:00:00', '07:00:00','14:00:00');**

**insert into unit(nume, adresa, idP)**

**values ('Somesana', 'Strada 1 nr.1', 1),**

**('Catena', 'Bvd. Bucuresti nr.1', 2),**

**('Policlinica 1', 'Str. Victoriei nr.1', 3);**

**insert into user1(username, parola, cnp, nume, prenume, adresa, numarTel, contIban, nrCon, dataAngajarii, functie, salariu, nrOre)**

**values('medic1','medic1',101,'Paul','Stefanescu','Strada nr 3','0766123123','50BTRL..', 2, '2020-11-11', 'medic', 2000, 180),**

**('medic2','medic2',102,'Raul','Dumitrescu','Strada nr 4','0762223123','75BTRL..', 3, '2019-1-2', 'medic', 2500, 190),**

**('medic3','medic3',103,'Anghel','Pop','Strada nr 5','0722223123','80BTRL..', 4, '2018-1-2', 'medic', 3000, 200),**

**('asist1','asist1',104,'Ana','Mateescu','Strada nr 6','07999999999','60BTRL..', 5, '2019-10-20', 'asistent', 2000, 200),**

**('asist2','asist2',105,'Andrei','Ioan','Strada nr 7','07888888888','12BTRL..', 6, '2019-10-21', 'asistent', 2000, 200),**

**('asist3','asist3',106,'Maria','Pop','Strada nr 8','07776543212','22BTRL..', 7, '2019-10-22', 'asistent', 2000, 200);**

**insert into asistent(CNP,tipul,grad)**

**values(104,'asistent general',1),**

**(105,'generalist',2),**

**(106,'pediatrie',2);**

**insert into medic(CNP,codParafa ,titlu ,post ,procent)**

**values(101,'1','doctorand','conferentiar',10),**

**(102,'2','masterand','conferentiar',15),**

**(103,'3','doctorand','conferentiar',20);**

**insert into cabinet(specialitate ,unit)**

**values('cardiologie', 'Catena'),**

**('pediatrie', 'Somesana'),**

**('chirurgie', 'Policlinica 1');**

**set @flag = null;**

**call addServiciu(101, 'RMN','Ecografie',100,2,@flag);**

**call addServiciu(102, 'Chirurgie','Chirurgie',1000,4,@flag);**

**call addServiciu(103, 'Consult general','Consult',50,1,@flag);**

**insert into Orare\_generice(CNP ,zi , ora\_inceput ,ora\_sfarsit, unitate\_medicala)**

**values(101,'luni', '07:00:00','14:00:00', 'Catena'),**

**(101,'marti', '07:00:00','14:00:00', 'Catena'),**

**(101,'miercuri', '07:00:00','14:00:00', 'Catena'),**

**(101,'joi', '07:00:00','14:00:00', 'Catena'),**

**(101,'vineri', '07:00:00','14:00:00', 'Catena'),**

**(101,'sambata', '07:00:00','14:00:00', 'Catena'),**

**(101,'duminica', '07:00:00','14:00:00', 'Catena'),**

**#primul medic**

**(102,'luni', '07:00:00','14:00:00', 'Somesana'),**

**(102,'marti', '07:00:00','14:00:00', 'Somesana'),**

**(102,'miercuri', '07:00:00','14:00:00', 'Somesana'),**

**(102,'joi', '07:00:00','14:00:00', 'Somesana'),**

**(102,'vineri', '07:00:00','14:00:00', 'Somesana'),**

**(102,'sambata', '07:00:00','14:00:00', 'Somesana'),**

**(102,'duminica', '07:00:00','14:00:00', 'Somesana'),**

**#al doilea medic**

**(103,'luni', '07:00:00','14:00:00', 'Policlinica 1'),**

**(103,'marti', '07:00:00','14:00:00', 'Policlinica 1'),**

**(103,'miercuri', '07:00:00','14:00:00', 'Policlinica 1'),**

**(103,'joi', '07:00:00','14:00:00', 'Policlinica 1'),**

**(103,'vineri', '07:00:00','14:00:00', 'Policlinica 1'),**

**(103,'sambata', '07:00:00','14:00:00', 'Policlinica 1'),**

**(103,'duminica', '07:00:00','14:00:00', 'Policlinica 1');**

**set @s:=null;**

**call addProgramare('Paul', 'Pop', 'Vasile', 101, 'RMN','Somesana', '09:00:00', '00:20:00', '2020-11-12',@s);**

**call addProgramare('Paul', 'Pop', 'Maria', 101, 'RMN','Somesana', '10:00:00', '00:20:00', '2020-09-10',@s);**

**call addProgramare('Raul', 'Ioan', 'Andrei', 102, 'Chirurgie','Catena', '12:00:00', '02:00:00', '2020-05-12',@s);**

**call addProgramare('Raul', 'Victoria', 'Mariana', 102, 'RMN','Catena', '11:00:00', '01:0:00', '2020-06-12',@s);**

**call addProgramare('Anghel', 'Marian', 'Ion', 103, 'Chirurgie','Policlinica 1', '08:00:00', '00:30:00', '2020-12-12',@s);**

**call addProgramare('Anghel', 'Popescu', 'Iordache', 103, 'Operatie de Menisc','Policlinica 1', '09:00:00', '03:00:00', '2020-01-25',@s);**

**insert into concediu(CNP, inceputConcediu, finalConcediu)**

**values(101, '2023-08-10','2023-08-24'),**

**(102, '2023-06-10','2023-06-24'),**

**(103, '2023-07-10','2023-07-24'),**

**(104, '2020-07-10','2020-07-24'),**

**(105, '2022-07-10','2022-07-24'),**

**(106, '2024-07-10','2024-07-24');**

**set @salar := null;**

**call calcSalar(101,'Paul',@salar);**

1. **Lista tabele**

**create database if not exists unitate\_medic;**

**use unitate\_medic;**

**create table if not exists unit(**

**nume varchar(255) not null primary key,**

**adresa varchar(255) not null,**

**idP int not null**

**);**

**create table if not exists user1(**

**username varchar(255) not null,**

**parola varchar(255) not null,**

**idU int,**

**cnp int(20) primary key not null,**

**nume varchar(255) not null,**

**prenume varchar(255) not null,**

**adresa varchar(255),**

**numarTel varchar(20),**

**contIban varchar(255),**

**nrCon int(20) not null,**

**dataAngajarii date not null,**

**functie varchar(30),**

**salariu int not null,**

**nrOre int not null**

**);**

**alter table user1**

**drop idU;**

**create table if not exists Orare\_generice**

**(id\_orar int primary key auto\_increment,**

**CNP int(20) not null,**

**zi varchar(100) not null,**

**ora\_inceput time not null,**

**ora\_sfarsit time not null,**

**unitate\_medicala varchar(100) not null,**

**foreign key(CNP) references user1(cnp)**

**);**

**create table if not exists Program**

**(id\_program int auto\_increment primary key,**

**luni\_inceput time not null,**

**luni\_sfarsit time not null,**

**marti\_inceput time not null,**

**marti\_sfarsit time not null,**

**miercuri\_inceput time not null,**

**miercuri\_sfarsit time not null,**

**joi\_inceput time not null,**

**joi\_sfarsit time not null,**

**vineri\_inceput time not null,**

**vineri\_sfarsit time not null,**

**sambata\_inceput time not null,**

**sambata\_sfarsit time not null,**

**duminica\_inceput time not null,**

**duminica\_sfarsit time not null**

**);**

**alter table unit**

**add foreign key (idP) references Program(id\_program);**

**create table if not exists medic(**

**CNP int(20) not null,**

**foreign key(CNP) references user1(cnp),**

**gradul varchar(30),**

**codParafa varchar(30),**

**titlu varchar(100),**

**post varchar(100),**

**procent int not null**

**);**

**alter table medic**

**drop column gradul;**

**create table if not exists asistent(**

**CNP int(20) not null,**

**foreign key(CNP) references user1(CNP),**

**tipul varchar(30),**

**grad varchar(30)**

**);**

**create table if not exists concediu(**

**idC int primary key auto\_increment,**

**CNP int(20) not null,**

**inceputConcediu date,**

**finalConcediu date,**

**foreign key (CNP) references user1(cnp)**

**);**

**create table if not exists servicii(**

**idServiciu int auto\_increment primary key,**

**serviciu varchar(100),**

**specialitate varchar(100),**

**pret int,**

**durata int,**

**cnp int(20),**

**foreign key (cnp) references medic(cnp)**

**);**

**create table if not exists programare(**

**id\_p int primary key auto\_increment,**

**ziua date,**

**ora time,**

**durata time,**

**CNP int(20),**

**serviciu varchar(100),**

**unitate\_medicala varchar(255),**

**nume\_pacient varchar(255) not null,**

**prenume\_pacient varchar(255) not null,**

**FOREIGN KEY (CNP) REFERENCES medic(cnp)**

**);**

**create table if not exists cabinet**

**(id\_cabinet int primary key not null auto\_increment,**

**specialitate varchar(100) not null,**

**unit varchar(100) not null,**

**FOREIGN KEY (unit) REFERENCES unit(nume)**

**);**

**create table if not exists repartizari(**

**idRepartizari int primary key auto\_increment,**

**cnp int(20),**

**id\_cabinet int,**

**foreign key (cnp) references medic(CNP),**

**foreign key (cnp) references cabinet(id\_cabinet)**

**);**

1. **Manual de functionare**

**Interfata va porni cu o pagina in care alegi policlinica**

Logo, company name

Description automatically generated

**Dupa ce policlinica este aleasa se va deschide meniul de logare unde va trebui sa introducem username-ul si parola corecta.**

**In partea de jos avem meniul de Sign In care ne introduce in baza de date.**

A picture containing logo

Description automatically generated

Graphical user interface, application

Description automatically generated

**Pentru fiecare rol avem un meniu diferit. Acestea se pot vedea inserand username-ul si parola curenta.**