A picture containing text, clipart

Description automatically generated Universitatea "Dunărea de Jos" din Galați

A picture containing text, clipart

Description automatically generatedFacultatea de Automatică, Calculatoare, Inginerie Electrică și Electronică

**Proiectarea Aplicațiilor Orientate pe Obiecte**

**- proiect-**

Coordonator: Student:

Șl. dr. ing. Cristina Elena ANTON George – Marius BARANGA

Grupa : 22C31A

Specializare : Calculatoare

Gestiunea unui magazin de instrumente muzicale

Analiza problemei

Un instrument muzical este un obiect utilizat în interpretarea unei compoziții [muzicale](https://ro.wikipedia.org/wiki/Muzic%C4%83).

Instrumentele muzicale se pot clasifica în funcție de numeroase criterii: calitatea sunetului emis, dificultatea folosirii, dificultatea studierii, principiul de funcționare, pregătirea muzicianului și așa mai departe. Un alt criteriu în funcție de care se face clasificare instrumentelor este greutatea acestora.

Din categoria instrumentelor cu coarde (coarde frecate cu arcuș, coarde ciupite sau coarde lovite):

* cele mai grele sunt: harpa, pianul acustic (sau orga), acordeonul, contrabasul, violoncelul, țambalul, chitara electrică;
* cele mai ușoare sunt: vioara, viola, mandolina, chitara acustică.

Din categoria instrumentelor de suflat:

* cele mai grele sunt: baritonul, saxofonul, trompeta, clarinetul;
* cele mai ușoare sunt: naiul, fluierul, flautul.

Din categoria instrumentelor de percuție:

* cele mai grele sunt: tobele electrice și cele clasice;
* cele mai ușoare sunt: tamburina și cinelele.

Angajati:

* **Casierul**
* Tine inventarul achizitiilor prin inregistrarea preturilor produselor intr-un registru de numerar;
* Introduce in baza de date modificarile de preturi si reducerile pentru produsele aflate la promotie;
* Ofera reduceri in baza voucherelor si cupoanelor detinute de clienti;
* Colecteaza platile prin acceptarea de numerar, bonuri valorice sau card si acorda restul in cazul platilor cash;
* Tine evidenta numerarului din casa, prin numararea acestuia la inceputul si la finalul turei;
* Ofera informatii despre preturi si raspunde intrebarilor clientilor;
* **Consilier vanzari**
* Directioneaza clientii prin escortarea lor la rafturi, sugerandu-le articolele de care au nevoie;
* Ii sfatuieste pe clienti oferindu-le informatii despre produsele existente pe rafturi;
* Ajuta clientii sa achizitioneze produse prin castigarea increderii acestora si oferindu-le sugestii;
* Tine evidenta vanzarilor prin crearea sau actualizarea profilului clientilor;
* Mentine clientela informata prin acordarea de informatii privind stocurile de produse si primirea de marfa;
* **Asistent de manager**
* Instruieste personalul magazinului si se asigura ca angajatii isi indeplinesc sarcinile de serviciu; desfasoara sesiuni de formare a personalului si tine legatura cu superiorul direct pentru a-l informa de evolutiile fiecarui angajat;
* Atrage clientii prin aranjarea marfii in magazin; urmeaza sugestiile sau orarele de afisare; produce afisaje pentru marfuri pe care le expune in ferestre si vitrine;
* Ajuta clientii prin furnizarea de informatii; raspunde la intrebarile acestora; se asigura de finalizarea tranzactiilor de plata; pregateste marfurile pentru livrare;
* Pregateste rapoarte pentru angajati in functie de vanzarile efectuate de acestia, modul de lucru cu clientii si identificarea si investigarea plangerilor clientilor si a sugestiilor de servicii;
* Mentine inventarul magazinului verificand stocurile si anticipand cererea clientilor;
* **Managerul**
* Face inventarul produselor emergente si anticipeaza interesul cumparatorului;
* are rolul de a instrui personalul magazinului si se asigura ca angajatii isi indeplinesc task-urile.
* responsabil de refacerea stocurilor, prin comandarea de noi produse
* negociaza preturile la achizitionarea de marfa pentru magazin;
* se asigura de plasarea si expedierea comenzilor;
* verificarea chitantei;

**Gestionarea magazinului pas cu pas**

* Managerul aprovizioneaza magazinul cu instrumente muzicale de la furnizori atunci cand stocul e pe terminate sau la cererea clientului de noi produse.
* Asistenul de manager va expune cate un exemplar din fiecare tip de instrument impreuna cu caracteristicile sale,pret si garantie.
* Consilierul de vanzari va indruma clientii in functie de cerintele lor catre produsul dorit si le va oferi detalii despre acestea astfel incat sa i convinga sa il cumpere.
* Casierul va incheia procesul de vanzare, aplicand la pretul instrumentului eventualele cupoane de reducere, iar in cazul in care clientul doreste o garantie prelungita sa se adauge si pretul acesteia la totalul ce trebuie sa l achite prin metoda de plata selectata.

**Ce roluri va avea aplicatia?**

Managerul

* va putea expune noi produse , sa le modifice sau sa le elimine;
* va putea sa actualizeze profilul clientilor;
* va putea sa refaca stocurile de la furnizori;
* exportul si importul datelor
* generarea de grafice

Clientul

* va putea cauta un instrument dupa anumite caracteristici si va putea plasa comenzi;

**MCD:**

Graphical user interface, diagram, application

Description automatically generated

**MLD:**

USER[ id\_u,nume,username,password,nivel\_acces]

BON[id\_bon,modalitate\_plata, cod\_prod,nume\_prod,ora,data,id\_angajat]

STOC[id\_produs,id\_furnizor,cantitate]

PRODUS [id\_produs,categorie,denumire,greutate,garantie,pret]

FURNIZOR[id\_furnizor,denumire, adresa\_sediu]

**MFD:**

-- MySQL Workbench Forward Engineering

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

-- -----------------------------------------------------

-- Schema bgmusic

-- -----------------------------------------------------

DROP SCHEMA IF EXISTS `bgmusic` ;

-- -----------------------------------------------------

-- Schema bgmusic

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `bgmusic` DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci ;

SHOW WARNINGS;

USE `bgmusic` ;

-- -----------------------------------------------------

-- Table `USER`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `USER` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `USER` (

`id\_u` INT NOT NULL AUTO\_INCREMENT,

`nume` VARCHAR(45) NULL,

`username` VARCHAR(45) NULL,

`password` VARCHAR(45) NULL,

`nivel\_acces` INT NOT NULL,

PRIMARY KEY (`id\_u`))

ENGINE = InnoDB;

SHOW WARNINGS;

-- -----------------------------------------------------

-- Table `BON`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `BON` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `BON` (

`id\_bon` INT NOT NULL,

`modalitate\_plata` VARCHAR(45) NULL,

`cod\_prod` INT NULL,

`nume\_prod` VARCHAR(45) NULL,

`ora` VARCHAR(45) NULL,

`data` DATE NULL,

`USER\_id\_u` INT NOT NULL,

PRIMARY KEY (`id\_bon`))

ENGINE = InnoDB;

SHOW WARNINGS;

-- -----------------------------------------------------

-- Table `PRODUS`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `PRODUS` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `PRODUS` (

`id\_produs` INT NOT NULL,

`categorie` VARCHAR(45) NULL,

`denumire` VARCHAR(45) NULL,

`greutate` INT NULL,

`garantie` VARCHAR(45) NULL,

`pret` INT NULL,

PRIMARY KEY (`id\_produs`))

ENGINE = InnoDB;

SHOW WARNINGS;

-- -----------------------------------------------------

-- Table `FURNIZOR`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `FURNIZOR` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `FURNIZOR` (

`id\_furnizor` INT NOT NULL,

`denumire` VARCHAR(45) NULL,

`adresa\_sediu` VARCHAR(45) NULL,

PRIMARY KEY (`id\_furnizor`))

ENGINE = InnoDB;

SHOW WARNINGS;

-- -----------------------------------------------------

-- Table `STOC`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `STOC` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `STOC` (

`cantitate` INT NULL,

`PRODUS\_id\_produs` INT NOT NULL,

`FURNIZOR\_id\_furnizor` INT NOT NULL)

ENGINE = InnoDB;

SHOW WARNINGS;

-- -----------------------------------------------------

-- Table `PRODUS\_has\_BON`

-- -----------------------------------------------------

DROP TABLE IF EXISTS `PRODUS\_has\_BON` ;

SHOW WARNINGS;

CREATE TABLE IF NOT EXISTS `PRODUS\_has\_BON` (

`PRODUS\_id\_produs` INT NOT NULL,

`BON\_id\_bon` INT NOT NULL,

PRIMARY KEY (`PRODUS\_id\_produs`, `BON\_id\_bon`))

ENGINE = InnoDB;

SHOW WARNINGS;

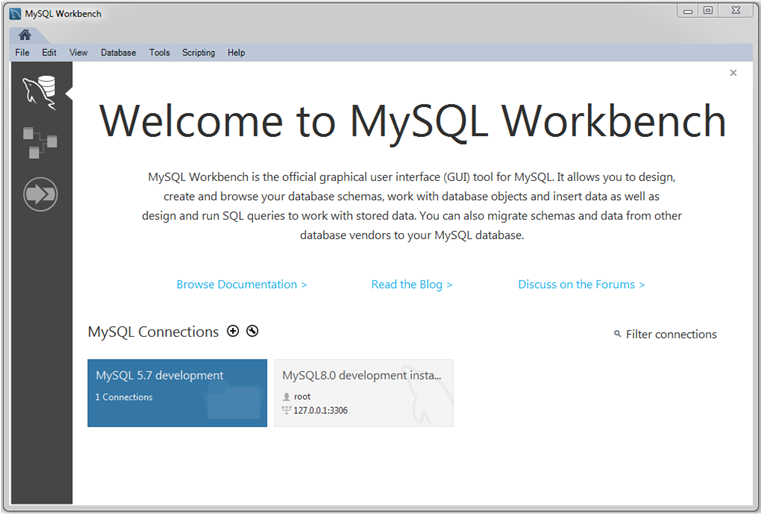
SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

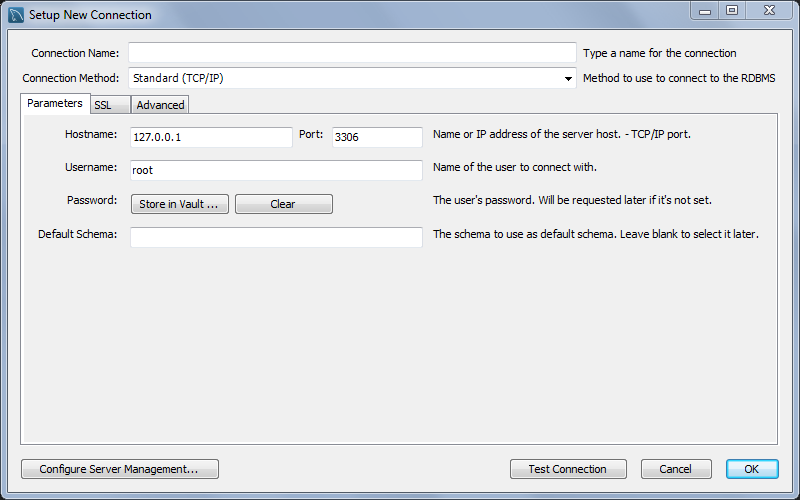
SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

1. Tehnologii si librarii folosite
   1. MySQL
      1. MySQL este un sistem de gestiune a bazelor de date relaționale, produs de compania suedeza MySQL AB și distribuit sub Licența Publică Generală GNU. Este cel mai popular SGBD open-source la ora actuală[6], fiind o componentă cheie a stivei LAMP (Linux, Apache, MySQL, PHP).

După cum este prezentat în figura următoare, un panou lateral al ecranului de pornire vă permite să comutați între Conexiunile MySQL (selectate în figură) și Modele din fila de pornire. Ultima opțiune din panoul lateral deschide MySQL Workbench Migration Wizard într-o filă nouă



Pentru a adăuga o conexiune, faceți clic pe [+]pictograma din dreapta titlului MySQL Connections de pe ecranul de [start](https://dev.mysql.com/doc/workbench/en/wb-home.html) . Aceasta deschide formularul Configurare conexiune nouă , după cum arată figura următoare.



* 1. C#
     1. contine mai multe facilitati novatoare, dintre care cele mai importante se refera la suportul incorporat pentru componente software. C# dispune de facilitati care implementeaza direct elementele care alcatuiesc componentele software, cum ar fi proprietatile, metodele si evenimentele. Poate cea mai importanta facilitate de caredispune C# este posibilitatea de a lucra intr-un mediu cu limbaj mixt

DESCRIEREA APLICATIEI:

Pagina principală:

* + Pagină de login + pagină de register(cont existent – user:admin;pass:admin)

Graphical user interface

Description automatically generated Graphical user interface, application

Description automatically generated

Conturile sunt create pe trei nivele:

* + Admin(0)
  + Angajat(1)
  + User obișnuit/client(2), pe baza câmpului nivel\_acces.

Pagina adminului: (principală)

Graphical user interface, text, application

Description automatically generated

* + Filtrare dupa pret;
  + Cautare dupa denumire;
  + Stergere si actualizare dupa codul produsului(scris sau selectat din datagridview)
  + Export(xls,csv) si import(csv).

Adaugă produs + actualizare produs:

Graphical user interface, text, application

Description automatically generated

Facilități:

* + Adaugă produs;
  + Actualizează produs.

Grafic:

* + Putem genera un grafic referitor la numărul de produse pentru fiecare categorie.

Chart, bar chart

Description automatically generated

Aplicația este în curs de dezvoltare.