

Proiect BD2

- Sistem de gestionare a elevilor, profesorilor si specializarilor -
 - unui liceu -

Student: Calota Alexandra Maria

Grupa: 341C3

Descrierea temei	3
Descrierea bazei de date	4
Diagrama bazei de date	4
Structura tabelelor	4
Descrierea constrângerilor de integritate	5
Descrierea procedurilor si funcțiilor	5
Descrierea aplicației	6
Diagrama de clase	6
Structura claselor	6
Diagrama de stări și fluxul de lucru (workflow) pentru aplicație;	6
Prezentarea modului în care se face conexiunea cu baza de date	6
Capturi de ecran pentru interfețe și rapoarte	7
Concluzii	10
Bibliografie	10

Descrierea temei

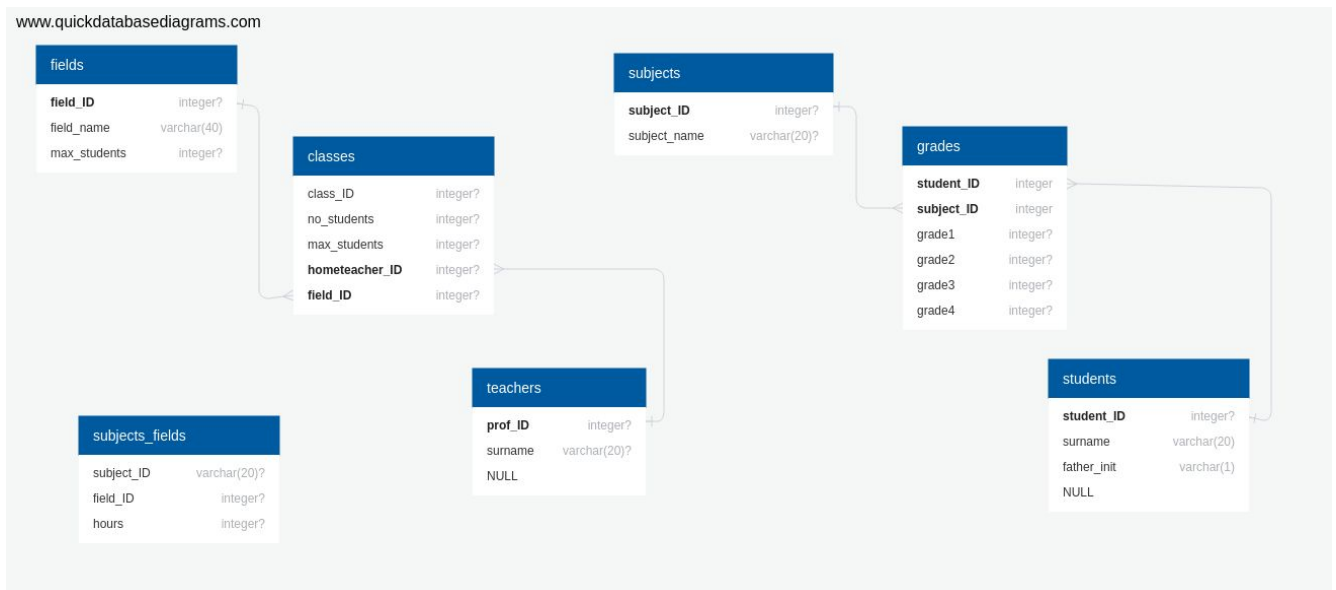
Am implementat o aplicatie pentru administrarea elevilor, profesorilor, specializarilor, claselor si notelor de la un colegiu cu clasele 11 si 12.

Am folosit docker-compose, am creat 2 containere - unul pentru server si unul pentru baza de date.

Pentru implementare, am folosit mysql si python.

Descrierea bazei de date

1. Diagrama bazei de date



2. Structura tabelelor

Table fields:

- field_ID integer auto_increment primary key,
- field_name varchar(40) unique not null,
- max_students integer

Table subjects:

- subject_ID integer auto_increment primary key,
- subject_name varchar(20)

Table subjects_fields:

- subject_ID varchar(20),
- field_ID integer,
- hours integer

Table teachers:

- prof_ID integer auto_increment primary key,
- surname varchar(20),
- name varchar(20),
- subject_ID integer not null,
- hiredate date,
- FOREIGN KEY (subject_ID) REFERENCES subjects(subject_ID)

Table classes:

- class_ID integer auto_increment primary key,

- no_students integer,
- max_students integer,
- hometeacher_ID integer,
- field_ID integer,
- year integer,
- FOREIGN KEY (field_ID) REFERENCES fields(field_ID),
- FOREIGN KEY (hometeacher_ID) REFERENCES teachers(prof_ID)

Table students:

- student_ID integer primary key auto_increment,
- surname varchar(20) not null,
- father_init varchar(1) not null,
- name varchar(20) not null,
- CNP varchar(20) unique,
- class_ID integer,
- FOREIGN KEY (class_ID) REFERENCES classes(class_ID)

Table grades:

- student_ID integer not null,
- subject_ID integer not null,
- grade1 integer DEFAULT 0,
- grade2 integer DEFAULT 0,
- grade3 integer DEFAULT 0,
- grade4 integer DEFAULT 0,
- FOREIGN KEY (subject_ID) REFERENCES subjects(subject_ID),
- FOREIGN KEY (student_ID) REFERENCES students(student_ID)

- Descrierea constrângerilor de integritate

S-a asigurat faptul ca id-urile profesorilor, studentilor, claselor si specializarilor sunt primary keys. S-a asigurat legatura intre tabele prin definirea foreign key-urilor.

- Descrierea procedurilor si funcțiilor

- Functia get_grades(surname, father_init, name, subjec) returns grades

Returneaza notele obtinute de un elev la o anumita materie.

- Procedura add_grade(IN surname, IN father_init, IN name, IN subject, IN grade)

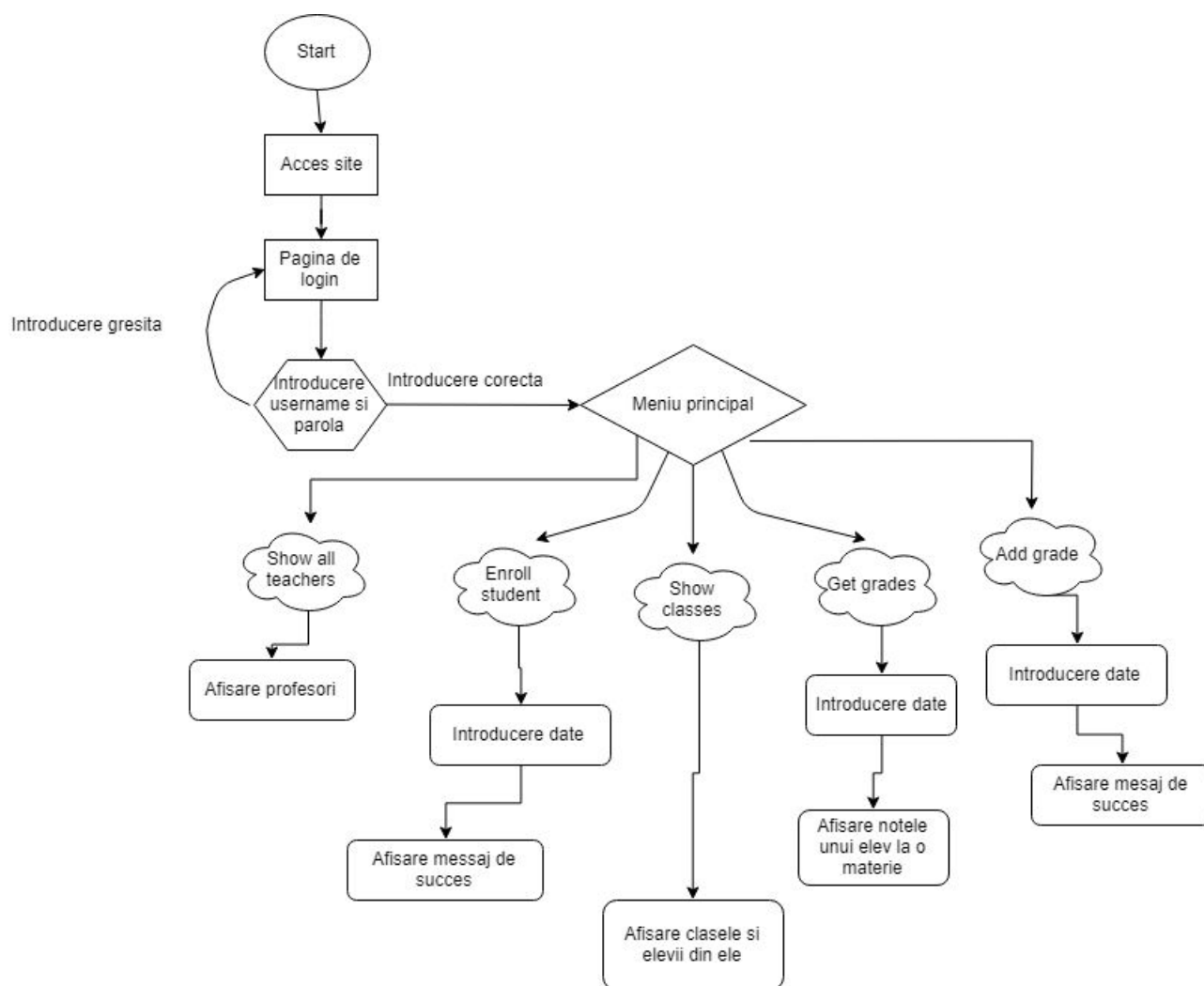
Adauga o nota pentru un anumit student, la o anumita materie.

- Procedura enroll_student(IN surname, IN father_init, IN name, IN CNP, IN field, IN year)

Inroleaza un student la liceu. Creste numarul de elevi din clasa si adauga elevul in tabela de elevi. Ii adauga de asemenea intrari in tabela de note.

Descrierea aplicației

1. Diagrama de stări și fluxul de lucru (workflow) pentru aplicație



2. Prezentarea modului în care se face conexiunea cu baza de date

Cu ajutorul docker-compose, au fost create 2 containere, unul pentru baza de date, ce folosește imaginea mysql, și unul pentru server. Clientul este reprezentat de browser, care accesează adresa <http://localhost:5000/> și urmează pașii prezentați. Serverul tratează aceste cereri http cu ajutorul pachetului flask.

Conexiunea serverului la baza de date se face cu ajutorul pachetului mysql.connector din python. Se creează un obiect de tip cursor, ale cărui metode 'execute' și 'callproc' se apelează pentru a efectua operații pe baza de date.



High School Database

Username:

Password:



High School Database

Menu

- ☐ Show all teachers
- ☒ Show classes
- ☐ Enroll student
- ☐ Add grade
- ☐ Get grades



High School Database

Enroll student

Surname

Father's name initial

Name

CNP

Year

Field of study



High School Database

Add Grade

Surname

Father's name's initial

Name

Subject

Grade



High School Database

Show Grades

Surname

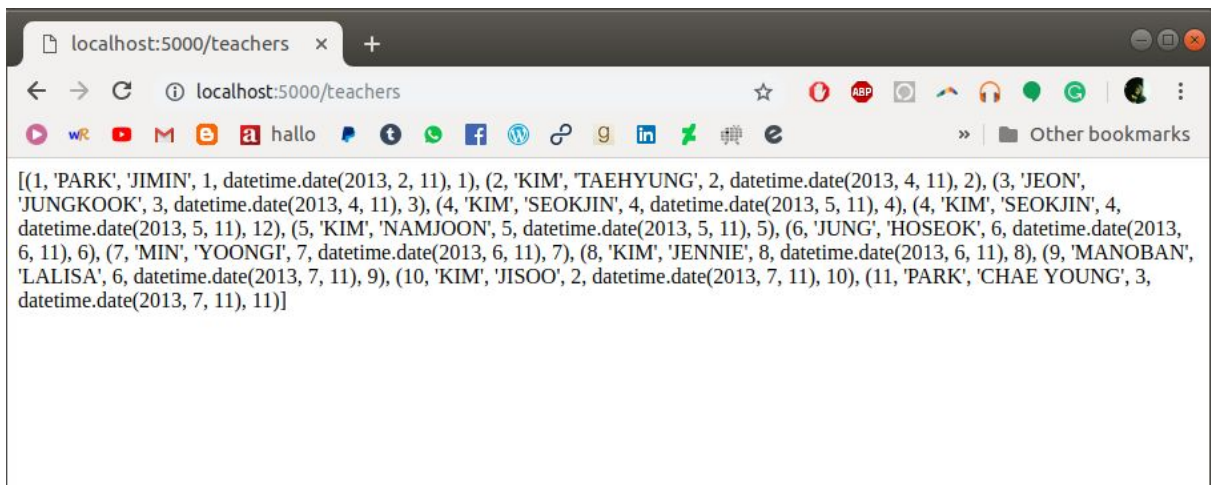
Father's name's initial

Name

Subject

Mathematics ▾

Get grades



[(1, 'Michael', 'Jackson'), (1, 'Bradley', 'Cooper'), (1, 'Rachel', 'McAddams'), (1, 'Nina', 'Dobrev'), (1, 'Paul', 'Weasley'), (1, 'Jules', 'Hugh'), (1, 'Brooks', 'Laich'), (1, 'Lady', 'Gaga'), (2, 'Michael', 'Jackson'), (2, 'Bradley', 'Cooper'), (2, 'Rachel', 'McAddams'), (2, 'Nina', 'Dobrev'), (2, 'Paul', 'Weasley'), (2, 'Jules', 'Hugh'), (2, 'Brooks', 'Laich'), (2, 'Lady', 'Gaga'), (3, 'Michael', 'Jackson'), (3, 'Bradley', 'Cooper'), (3, 'Rachel', 'McAddams'), (3, 'Nina', 'Dobrev'), (3, 'Paul', 'Weasley'), (3, 'Jules', 'Hugh'), (3, 'Brooks', 'Laich'), (3, 'Lady', 'Gaga'), (4, 'Michael', 'Jackson'), (4, 'Bradley', 'Cooper'), (4, 'Rachel', 'McAddams'), (4, 'Nina', 'Dobrev'), (4, 'Paul', 'Weasley'), (4, 'Jules', 'Hugh'), (4, 'Brooks', 'Laich'), (4, 'Lady', 'Gaga'), (5, 'Michael', 'Jackson'), (5, 'Bradley', 'Cooper'), (5, 'Rachel', 'McAddams'), (5, 'Nina', 'Dobrev'), (5, 'Paul', 'Weasley'), (5, 'Jules', 'Hugh'), (5, 'Brooks', 'Laich'), (5, 'Lady', 'Gaga'), (6, 'Michael', 'Jackson'), (6, 'Bradley', 'Cooper'), (6, 'Rachel', 'McAddams'), (6, 'Nina', 'Dobrev'), (6, 'Paul', 'Weasley'), (6, 'Jules', 'Hugh'), (6, 'Brooks', 'Laich'), (6, 'Lady', 'Gaga'), (7, 'Michael', 'Jackson'), (7, 'Bradley', 'Cooper'), (7, 'Rachel', 'McAddams'), (7, 'Nina', 'Dobrev'), (7, 'Paul', 'Weasley'), (7, 'Jules', 'Hugh'), (7, 'Brooks', 'Laich'), (7, 'Lady', 'Gaga'), (8, 'Michael', 'Jackson'), (8, 'Bradley', 'Cooper'), (8, 'Rachel', 'McAddams'), (8, 'Nina', 'Dobrev'), (8, 'Paul', 'Weasley'), (8, 'Jules', 'Hugh'), (8, 'Brooks', 'Laich'), (8, 'Lady', 'Gaga'), (9, 'Michael', 'Jackson'), (9, 'Bradley', 'Cooper'), (9, 'Rachel', 'McAddams'), (9, 'Nina', 'Dobrev'), (9, 'Paul', 'Weasley'), (9, 'Jules', 'Hugh'), (9, 'Brooks', 'Laich'), (9, 'Lady', 'Gaga'), (10, 'Michael', 'Jackson'), (10, 'Bradley', 'Cooper'), (10, 'Rachel', 'McAddams'), (10, 'Nina', 'Dobrev'), (10, 'Paul', 'Weasley'), (10, 'Jules', 'Hugh'), (10, 'Brooks', 'Laich'), (10, 'Lady', 'Gaga'), (11, 'Michael', 'Jackson'), (11, 'Bradley', 'Cooper'), (11, 'Rachel', 'McAddams'), (11, 'Nina', 'Dobrev'), (11, 'Paul', 'Weasley'), (11, 'Jules', 'Hugh'), (11, 'Brooks', 'Laich'), (11, 'Lady', 'Gaga'), (12, 'Michael', 'Jackson'), (12, 'Bradley', 'Cooper'), (12, 'Rachel', 'McAddams'), (12, 'Nina', 'Dobrev'), (12, 'Paul', 'Weasley'), (12, 'Jules', 'Hugh'), (12, 'Brooks', 'Laich'), (12, 'Lady', 'Gaga')]

Bibliografie

1. <https://docs.docker.com/>
2. <https://dev.mysql.com/>
3. <https://docs.python.org/3/>