UNIVERSITATEA TEHNICĂ “GH ASACHI” IAŞI
FACULTATEA DE AUTOMATICĂ ŞI CALCULATOARE
SECȚIA CALCULATOARE ŞI TEHNOLOGIA INFORMAŢIEI
SPECIALIZAREA CALCULATOARE
DISCIPLINA BAZE DE DATE PROIECT

Aplicatie de gestiune a terenurilor agricole

Coordonator,

Asist.drd.ing. Cătălin Mironeanu

Student,

Borcilă Petru-Emanuel
Grupa 1307A

Iași,2019

Tema proiectului

Proiectul isi propune proiectarea si realizarea unei baze de date care sa modeleze gestiunea terenurilor agricole pe care un utilizator (client) le are in posesie la un moment dat. De asemenea, aplicatia va gestiona si angajatii cu care clientul isi indeplineste activitatile cat si a utilajelor agricole pe care angajatii le utilizeaza.

Informatiile de care avem nevoie in cadrul unei asemenea aplicatii sunt cele legate de:

-utilizator sau client, despre care este necesar sa stim un set de detalii ce cuprind numele, e-mail-ul numarul de telefon si parola aferenta contului care se va realiza in cadrul aplicatiei

-angajatii clientului, despre care trebuie sa cunoastem detalii precum numele si salariul acestora

-vehiculele pe care clientul le are trimise pe camp, despre care trebuie sa cunoastem tipul vehiculului (e.g. tractor, combina, remorca etc) cat si statusul acestuia (in functiune, stricat, in garaj)

-terenurile agricole pe care clientul le detine, despre care trebuie sa cunoastem detalii precum lungimea terenului, latimea terenului, suprafata terenului si un numar de identificare al terenului spre a putea face o gestiune mult mai usoara a acestuia bazata pe caractere numerice carora li s-au atribuit dimensiunile terenului

Aceasta baza de date este una simpla, dar, in acelasi timp, foarte utila, putand fi folosita usor de catre orice utilizator prin intermediul unei aplicatii, ea fiind capabila sa faciliteze gestiunea activitatilor agricole prin eliminarea factorilor de eroare umani.

Descrierea detaliata a entitatilor si a relatiilor din tabele

In realizarea bazei de date s-au folosit urmatoarele tabele:

- ❖ Utilizator
- ❖ Detalii utilizator
- ❖ Angajati
- ❖ Terenuri
- ❖ Vehicule

In proiectarea acestei baze de date s-au identificat tipurile de relatii 1:1,1:n, si n:m.

Tabela users contine campurile user_id, email, parola. Intre tabela utilizator si tabela detalii_utilizator exista o relatie de 1:1, fiecare utilizator avand detaliile suplimentare aferente lui si numai lui in tabela de detalii.

Intre tabela utilizatori si angajati exista o relatie de 1:n, un utilizator poate avea in subordinea sa mai multi angajati.

Intre tabela vehicule si tabela angajati exista o relatie de m:n deoarece fiecare angajat poate utiliza in activitatea lui mai multe utilaje agricole.

De asemenea, intre tabela utilizatori si tabela vehicule este o relatie de 1:n pentru ca fiecare utilizator al aplicatiei va putea gestiona un numar n de vehicule agricole.

Si, in final, intre tabela utilizatori si tabela terenuri exista o relatie 1:n deoarece fiecare utilizator va putea gestiona un numar n de terenuri agricole.

Tabela utilizator contine un set minimal de detalii aferente fiecarui utilizator.

Tabela detalii utilizator contine setul extins de detale aferente fiecarui utilizator.

Tabela angajati contine setul de detalii aferent fiecarui angajat din subordinea unui utilizator.

Tabela vehicule contine vehicule contine campuri cu detalii despre vehicul, anume starea acestuia si tipul acestuia.

Tabela terenuri contine detaliile aferente fiecarui teren pe care un utilizator le are in posesie.

Relationarea dintre tabela utilizatori si tabela detalii_utilizator se face prin intermediul campului id_utilizator, acesta find un primary key in tabela utilizatori.

Relationarea dintre tabela utilizatori si tabela angajati se face tot prin intermediul campului id_utilizator.

Relationarea dintre campurile tabelelor vehicule si utilizatori se face prin intermediul aceluiasi id_utilizator.

Intre tabelele vehicule si angajati, campurile prin care se face relationarea sunt id_vehicul si id_angajat.

Pentru realizarea aplicatiei prin care utilizatorul are acces la baza de date s-au folosit urmatoarele tehnologii:

- front-end: HTML-bootstrap
- back-end: PHP

In realizarea bazei de date s-au folosit urmatoarele constrangeri:

-constrangere pentru ca lungimea numarului de telefon sa fie de exact zece cifre

-constrangere pentru ca statusul celor trei vehicule sa fie intre trei valori, anume, “defect”,”garaj” si “functional”.

-constrangere pentru ca lungimea parolei utilizatorului sa fie mai mare decat 6 caractere

Conectarea la baza de date s-a realizat prin crearea unui user, in prealabil, pe terminalul SQL command, iar apoi, prin conectarea cu acel user la baza de date, cu ajutorul codului PHP:

```
$db = oci_connect('emanuel', 'emanuel', 'localhost/XE');
if (!$db) {
    $e = oci_error();
    trigger_error(htmlentities($e['message'], ENT_QUOTES),
E_USER_ERROR);
}
```

Scriptul generat de Oracle Data Modeler impreuna cu, comenzile de inserare in baza de date:

```
-- Generated by
```

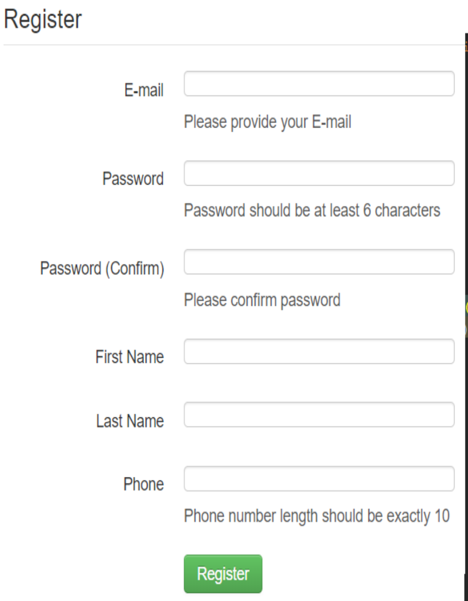
```
Oracle SQL
Developer Data
Modeler
18.4.0.339.1532
                  --   at:        2019-05-23 16:03:17 EEST
                  --   site:      Oracle Database 11g
                  --   type:      Oracle Database 11g



                  DROP TABLE employees CASCADE CONSTRAINTS;

                  DROP TABLE terrains CASCADE CONSTRAINTS;

                  DROP TABLE users CASCADE CONSTRAINTS;

                  DROP TABLE users_details CASCADE CONSTRAINTS;

                  DROP TABLE veh_emp_fk CASCADE CONSTRAINTS;

                  DROP TABLE vehicles CASCADE CONSTRAINTS;

                  CREATE TABLE employees (
                                             emp_id          NUMBER(3) NOT NULL,
                                             name            VARCHAR2(255) NOT NULL,
                                             salary          NUMBER(5) NOT NULL,
                                             users_user_id   NUMBER(3) NOT NULL
                  );

                  ALTER TABLE employees ADD CONSTRAINT employees_pk PRIMARY KEY ( emp_id );

                  ALTER TABLE employees ADD CONSTRAINT employees_emp_id_un UNIQUE ( emp_id
                  );

                  CREATE TABLE terrains (
                                            terrain_id      NUMBER(3) NOT NULL,
                                            area            NUMBER(6),
                                            width           NUMBER(3) NOT NULL,
                                            tlength         NUMBER(3) NOT NULL,
                                            users_user_id   NUMBER(3) NOT NULL
                  );

                  ALTER TABLE terrains ADD CONSTRAINT terrains_pk PRIMARY KEY ( terrain_id
);

ALTER TABLE terrains ADD CONSTRAINT terrains_terrain_id_un UNIQUE (
terrain_id );

CREATE TABLE users (
                       user_id   NUMBER(3) NOT NULL,
                       email     VARCHAR2(255) NOT NULL,
                       pwd       VARCHAR2(255) NOT NULL
);

ALTER TABLE users ADD CONSTRAINT users_pk PRIMARY KEY ( user_id );

ALTER TABLE users ADD CONSTRAINT users_user_id_un UNIQUE ( user_id );

ALTER TABLE users ADD CONSTRAINT users_email_un UNIQUE ( email );

ALTER TABLE users ADD CONSTRAINT users_pwd_ck CHECK (LENGTH(pwd) > 6 );

CREATE TABLE users_details (
                               first_name      VARCHAR2(255) NOT NULL,
                               last_name       VARCHAR2(255) NOT NULL,
                               phone           VARCHAR2(10),
                               users_user_id   NUMBER(3) NOT NULL
);

ALTER TABLE users_details ADD CONSTRAINT users_pwd_ck CHECK
(LENGTH(phone) = 6 );


CREATE UNIQUE INDEX users_details__idx ON
    users_details (
                   users_user_id
                   ASC );

CREATE TABLE veh_emp_fk (
                            vehicles_vehicle_id   NUMBER(3) NOT NULL,
                            employees_emp_id      NUMBER(3) NOT NULL
);

ALTER TABLE veh_emp_fk ADD CONSTRAINT veh_emp_fk_pk PRIMARY KEY (
vehicles_vehicle_id,
employees_emp_id );

CREATE TABLE vehicles (
                          vehicle_id      NUMBER(3) NOT NULL,
                          status          VARCHAR2(255) NOT NULL,
                          vehicle_type    VARCHAR2(255) NOT NULL,
                          users_user_id   NUMBER(3) NOT NULL
);

ALTER TABLE vehicles ADD CONSTRAINT vehicles_pk PRIMARY KEY ( vehicle_id
);

ALTER TABLE vehicles ADD CONSTRAINT vehicles_vehicle_id_un UNIQUE (
vehicle_id );

ALTER TABLE vehicles ADD CONSTRAINT veh_stat_ck CHECK (status IN (

'broken',

'garage',

'working'
    ));

ALTER TABLE employees
    ADD CONSTRAINT employees_users_fk FOREIGN KEY ( users_user_id )
        REFERENCES users ( user_id );

ALTER TABLE terrains
    ADD CONSTRAINT terrains_users_fk FOREIGN KEY ( users_user_id )
        REFERENCES users ( user_id );

ALTER TABLE users_details
    ADD CONSTRAINT users_details_users_fk FOREIGN KEY ( users_user_id )
        REFERENCES users ( user_id );

ALTER TABLE veh_emp_fk
    ADD CONSTRAINT veh_emp_fk_employees_fk FOREIGN KEY ( employees_emp_id
)
        REFERENCES employees ( emp_id );

ALTER TABLE veh_emp_fk
    ADD CONSTRAINT veh_emp_fk_vehicles_fk FOREIGN KEY (
vehicles_vehicle_id )
        REFERENCES vehicles ( vehicle_id );

ALTER TABLE vehicles
    ADD CONSTRAINT vehicles_users_fk FOREIGN KEY ( users_user_id )
        REFERENCES users ( user_id );

DROP SEQUENCE employees_emp_id_seq;
CREATE SEQUENCE employees_emp_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER employees_emp_id_trg BEFORE
    INSERT ON employees
    FOR EACH ROW
    WHEN ( new.emp_id IS NULL )
BEGIN
    :new.emp_id := employees_emp_id_seq.nextval;
END;
/

DROP SEQUENCE terrains_terrain_id_seq;
CREATE SEQUENCE terrains_terrain_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER terrains_terrain_id_trg BEFORE
    INSERT ON terrains
    FOR EACH ROW
    WHEN ( new.terrain_id IS NULL )
BEGIN
    :new.terrain_id := terrains_terrain_id_seq.nextval;
END;
/

DROP SEQUENCE users_user_id_seq;
CREATE SEQUENCE users_user_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER users_user_id_trg BEFORE
    INSERT ON users
    FOR EACH ROW
    WHEN ( new.user_id IS NULL )
BEGIN
    :new.user_id := users_user_id_seq.nextval;
END;
/
```

```
DROP SEQUENCE vehicles_vehicle_id_seq;
CREATE SEQUENCE vehicles_vehicle_id_seq START WITH 1 NOCACHE ORDER;

CREATE OR REPLACE TRIGGER vehicles_vehicle_id_trg BEFORE
    INSERT ON vehicles
    FOR EACH ROW
    WHEN ( new.vehicle_id IS NULL )
BEGIN
    :new.vehicle_id := vehicles_vehicle_id_seq.nextval;
END;
/

insert into users (email, pwd) values ('dgiannassi0@wp.com',
'KLxDynxM3');
insert into users (email, pwd) values ('holkowicz1@va.gov',
'x6HLg3WUQYP8');
insert into users (email, pwd) values ('asparshutt2@phpbb.com',
'GppHygby');
insert into users (email, pwd) values ('vrenner3@4shared.com',
'Du7r18IrTy');
insert into users (email, pwd) values ('eiwaszkiewicz4@xing.com',
'KAAWkWQ5OWS5');

insert into users_details (first_name, last_name, phone, users_user_id)
values ('Kingsly', 'McAmish', '2396174751',

(SELECT user_id from users WHERE email='dgiannassi0@wp.com')

);

insert into users_details (first_name, last_name, phone, users_user_id)
values ('Maureen', 'Tyreman', '5089876224',

(SELECT user_id from users WHERE email='holkowicz1@va.gov')

);

insert into users_details (first_name, last_name, phone, users_user_id)
values ('Lexis', 'Cheke', '2303302008',

(SELECT user_id from users WHERE email='asparshutt2@phpbb.com')

);
```

```
insert into users_details (first_name, last_name, phone, users_user_id)
values ('Alejandra', 'Burrow', '4946401885',

(SELECT user_id from users WHERE email='vrenner3@4shared.com')


);
insert into users_details (first_name, last_name, phone, users_user_id)
values ('Brittani', 'Walch', '6003736607',

(SELECT user_id from users WHERE email='eiwaszkiewicz4@xing.com')

);

insert into vehicles (status, vehicle_type, users_user_id) values
('garage', 'tractor',

(SELECT user_id from users WHERE email='dgiannassi0@wp.com')
                                                                  );

insert into vehicles (status, vehicle_type, users_user_id) values
('garage', 'tractor',

(SELECT user_id from users WHERE email='holkowicz1@va.gov')
                                                                  );

insert into vehicles (status, vehicle_type, users_user_id) values
('garage', 'tractor',

(SELECT user_id from users WHERE email='asparshutt2@phpbb.com')
                                                                  );

insert into vehicles (status, vehicle_type, users_user_id) values
('garage', 'tractor',

(SELECT user_id from users WHERE email='vrenner3@4shared.com')
                                                                  );

insert into vehicles (status, vehicle_type, users_user_id) values
('garage', 'tractor',

(SELECT user_id from users WHERE email='eiwaszkiewicz4@xing.com')
                                                                  );
```

```
insert into terrains (width, tlength, area, users_user_id) values (464,
531, (width*tlength),

(SELECT user_id from users WHERE email='dgiannassi0@wp.com')
                                                                  );

insert into terrains (width, tlength, area, users_user_id) values (102,
224, (102*224),

(SELECT user_id from users WHERE email='holkowicz1@va.gov')
                                                                  );

insert into terrains (width, tlength, area, users_user_id) values (430,
574, (430*574),

(SELECT user_id from users WHERE email='asparshutt2@phpbb.com')
                                                                  );

insert into terrains (width, tlength, area, users_user_id) values (793,
866, (793*866),

(SELECT user_id from users WHERE email='vrenner3@4shared.com')
                                                                  );

insert into terrains (width, tlength, area, users_user_id) values (884,
349, (884*349),

(SELECT user_id from users WHERE email='eiwaszkiewicz4@xing.com')
                                                                  );
insert into employees (name, salary, users_user_id) values ('Burr
Emerson',     734,
                                                            (SELECT
user_id from users WHERE email='dgiannassi0@wp.com')
                                                           );
insert into employees (name, salary, users_user_id) values ('Marc
Ranking',     755,
                                                            (SELECT
user_id from users WHERE email='holkowicz1@va.gov')
                                                           );
insert into employees (name, salary, users_user_id) values ('Gwenneth
Pecht',   739,
                                                            (SELECT
user_id from users WHERE email='asparshutt2@phpbb.com')
                                                          );
insert into employees (name, salary, users_user_id) values ('Roseanne
Sheppard',622,
                                                           (SELECT
user_id from users WHERE email='vrenner3@4shared.com')
                                                          );
insert into employees (name, salary, users_user_id) values ('Jessa
Bexley',     536,
                                                           (SELECT
user_id from users WHERE email='eiwaszkiewicz4@xing.com')
                                                          );
```

Register Page si codul in php



Register

E-mail

Please provide your E-mail

Password

Password should be at least 6 characters

Password (Confirm)

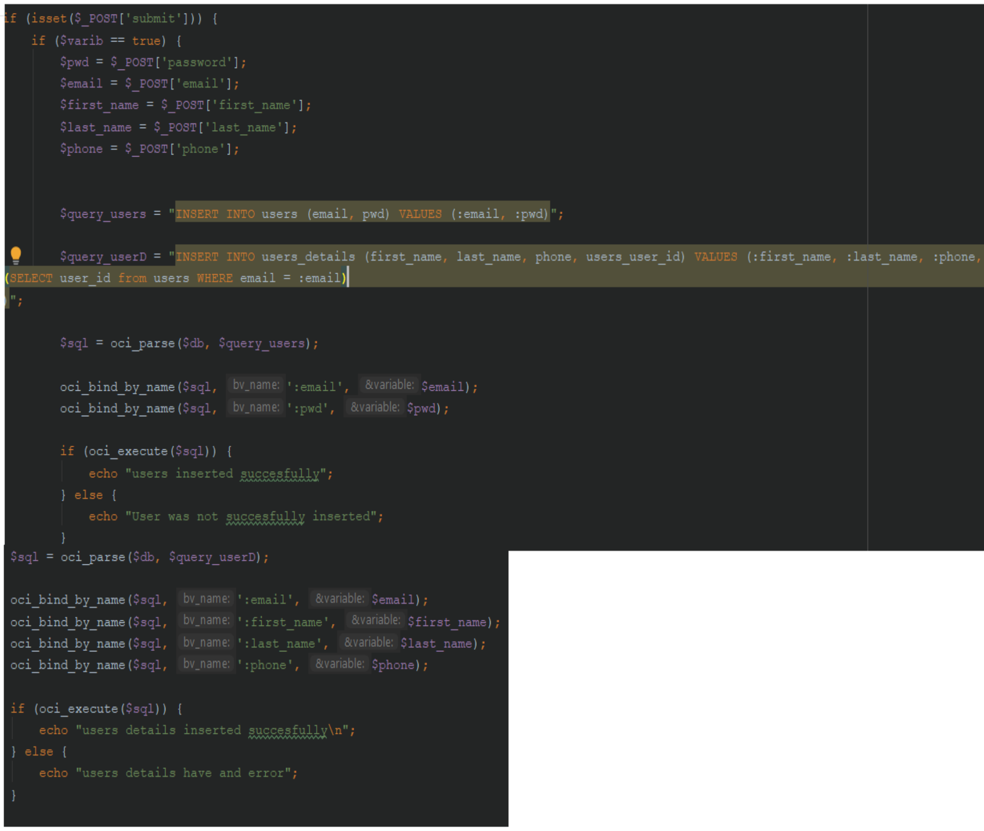
Please confirm password

First Name

Last Name

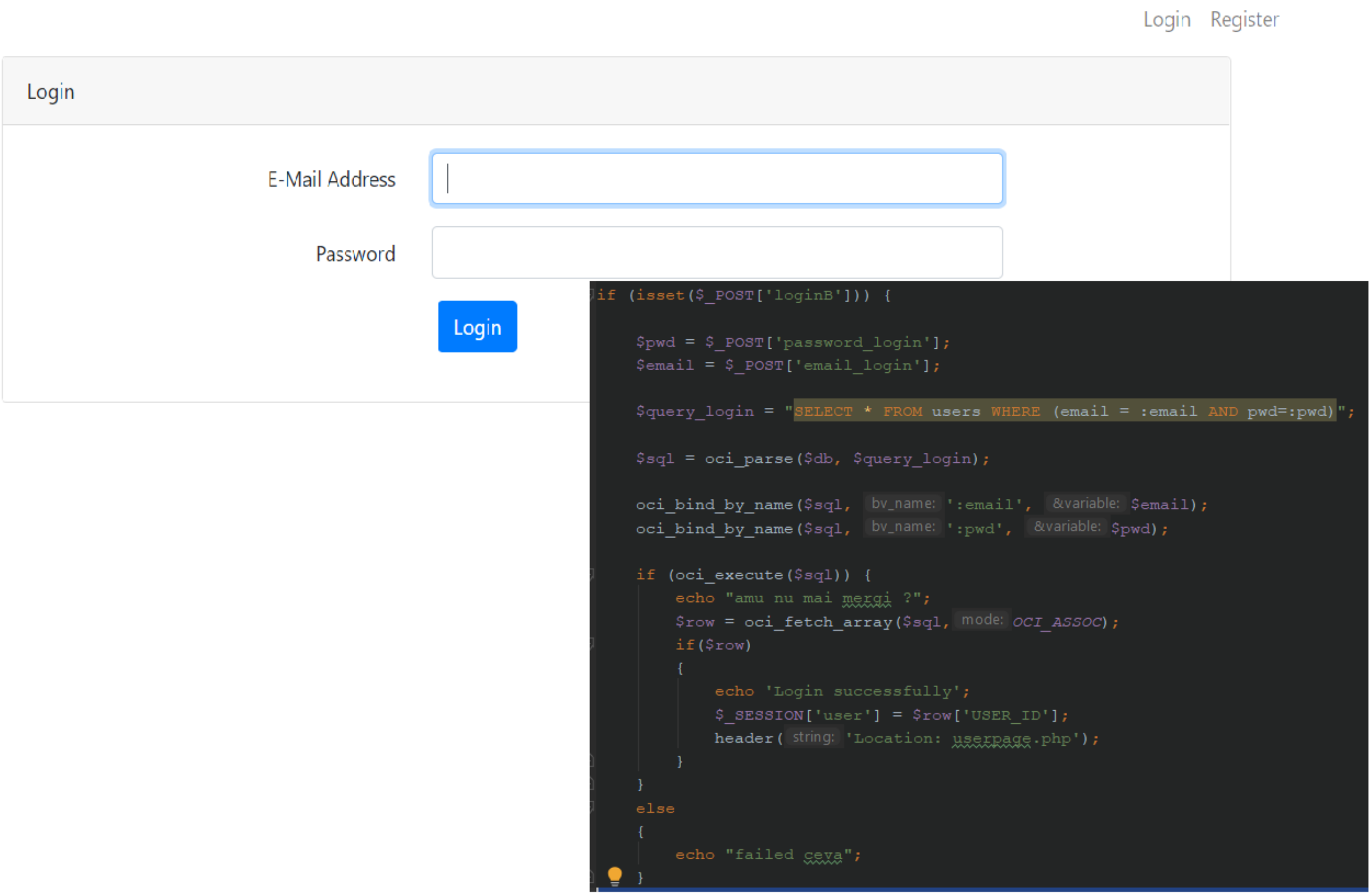
Phone

Phone number length should be exactly 10



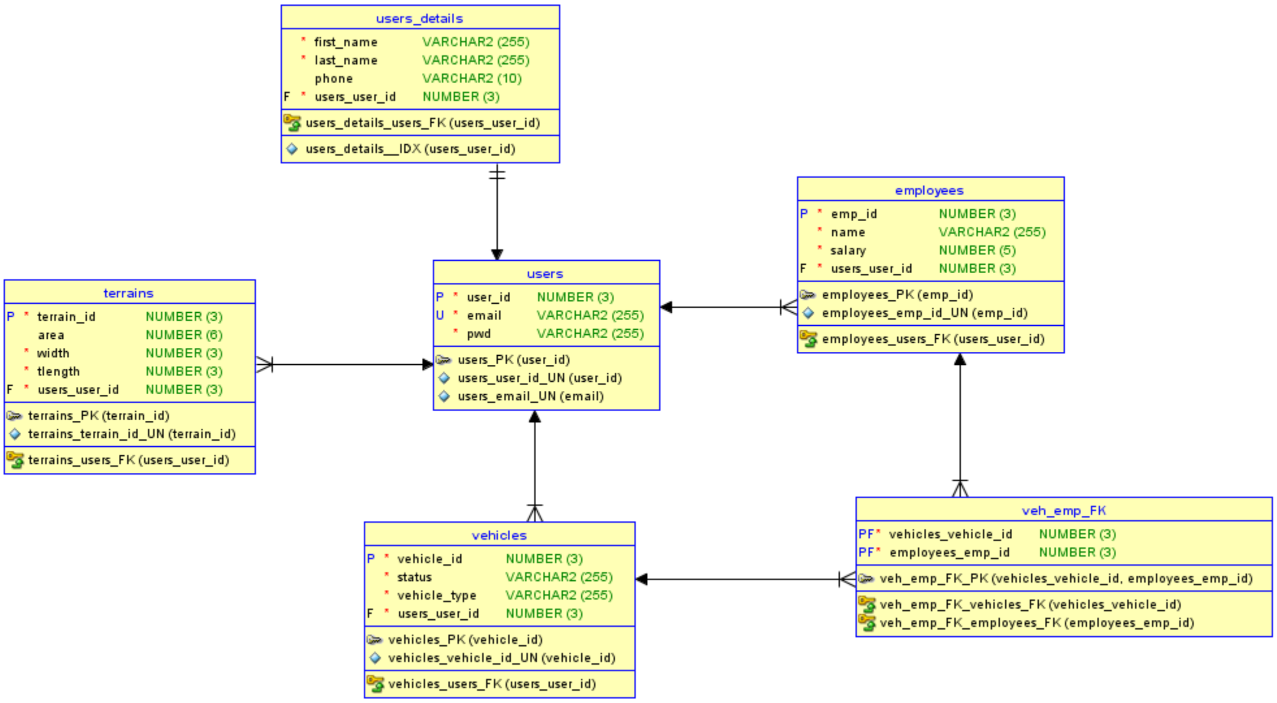
```
if (isset($_POST['submit'])) {
    if ($varib == true) {
        $pwd = $_POST['password'];
        $email = $_POST['email'];
        $first_name = $_POST['first_name'];
        $last_name = $_POST['last_name'];
        $phone = $_POST['phone'];


        $query_users = "INSERT INTO users (email, pwd) VALUES (:email, :pwd)";

        $query_userD = "INSERT INTO users_details (first_name, last_name, phone, users_user_id) VALUES (:first_name, :last_name, :phone,
(SELECT user_id from users WHERE email = :email)
)";

        $sql = oci_parse($db, $query_users);

        oci_bind_by_name($sql, bv_name: ':email', &variable: $email);
        oci_bind_by_name($sql, bv_name: ':pwd', &variable: $pwd);

        if (oci_execute($sql)) {
            echo "users inserted succesfully";
        } else {
            echo "User was not succesfully inserted";
        }
$sql = oci_parse($db, $query_userD);

oci_bind_by_name($sql, bv_name: ':email', &variable: $email);
oci_bind_by_name($sql, bv_name: ':first_name', &variable: $first_name);
oci_bind_by_name($sql, bv_name: ':last_name', &variable: $last_name);
oci_bind_by_name($sql, bv_name: ':phone', &variable: $phone);

if (oci_execute($sql)) {
    echo "users details inserted succesfully\n";
} else {
    echo "users details have and error";
}
```

Login Page si codul php



```
if (isset($_POST['loginB'])) {

    $pwd = $_POST['password_login'];
    $email = $_POST['email_login'];

    $query_login = "SELECT * FROM users WHERE (email = :email AND pwd=:pwd)";

    $sql = oci_parse($db, $query_login);

    oci_bind_by_name($sql, bv_name: ':email', &variable: $email);
    oci_bind_by_name($sql, bv_name: ':pwd', &variable: $pwd);

    if (oci_execute($sql)) {
        echo "amu nu mai mergi ?";
        $row = oci_fetch_array($sql, mode: OCI_ASSOC);
        if($row)
        {
            echo 'Login successfully';
            $_SESSION['user'] = $row['USER_ID'];
            header( string: 'Location: userpage.php');
        }
    }
    else
    {
        echo "failed ceva";
    }
```

Tabela Entitate-Relatie


users_details
* first_name VARCHAR2 (255)
* last_name VARCHAR2 (255)
phone VARCHAR2 (10)
F * users_user_id NUMBER (3)
users_details_users_FK (users_user_id)
users_details__IDX (users_user_id)
employees
P * emp_id NUMBER (3)
* name VARCHAR2 (255)
* salary NUMBER (5)
F * users_user_id NUMBER (3)
employees_PK (emp_id)
employees_emp_id_UN (emp_id)
employees_users_FK (users_user_id)
terrains
P * terrain_id NUMBER (3)
area NUMBER (6)
* width NUMBER (3)
* tlength NUMBER (3)
F * users_user_id NUMBER (3)
terrains_PK (terrain_id)
terrains_terrain_id_UN (terrain_id)
terrains_users_FK (users_user_id)
users
P * user_id NUMBER (3)
U * email VARCHAR2 (255)
* pwd VARCHAR2 (255)
users_PK (user_id)
users_user_id_UN (user_id)
users_email_UN (email)
veh_emp_FK
PF* vehicles_vehicle_id NUMBER (3)
PF* employees_emp_id NUMBER (3)
veh_emp_FK_PK (vehicles_vehicle_id, employees_emp_id)
veh_emp_FK_vehicles_FK (vehicles_vehicle_id)
veh_emp_FK_employees_FK (employees_emp_id)
vehicles
P * vehicle_id NUMBER (3)
* status VARCHAR2 (255)
* vehicle_type VARCHAR2 (255)
F * users_user_id NUMBER (3)
vehicles_PK (vehicle_id)
vehicles_vehicle_id_UN (vehicle_id)
vehicles_users_FK (users_user_id)