**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 graj)

-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 sase

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*);  
}