



**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE  
CATEDRA CALCULATOARE**

# **PROIECT**

la disciplina

Introducere in Baze de Date

## **Sistem de gestiune a unui centru de donare sânge**

**Juravle Monica-Nicoleta și Bîzgă Ana-Maria**

**An academic :2023 – 2024**

**Grupa: 30221**

PROIECT de SEMESTRU  
Catedra de Calculatoare  
Disciplina : Introducere in Baze de Date  
Coordonator: s.l. ing. Cosmina IVAN  
Data 17.01.2023



**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE  
CATEDRA CALCULATOARE**

## **Cuprins**

### **1. Introducere**

### **2. Analiza cerințelor utilizatorilor**

### **3. Modelul de date și descrierea acestuia**

### **4. Detalii de implementare**

### **5. Concluzii, limitări și dezvoltări ulterioare**

### **6. Bibliografie**



## **1. Introducere:**

Donarea de sânge reprezintă un act altruist și vital care salvează vieți și îmbunătățește starea de sănătate a celor care se confruntă cu diverse afecțiuni medicale. Fiecare donare oferă o șansă la viață pentru pacienți în nevoie, contribuind la tratarea unor afecțiuni grave, intervenții chirurgicale complexe și tratamentul bolilor cronice. Noi am ales această temă datorită apariției tot mai des a acestei probleme în mediul online și nu numai. Atât la televizor, cât și pe internet observăm o creștere a campaniilor care încurajează donarea de sânge pentru a salva vieți. Unii o fac din bunătate și spirit de solidaritate, alții datorită avantajelor ulterioare. Indiferent de motive, această aplicație a fost creată de noi două cu scopul de a facilita procesul de înregistrare, permițând oamenilor să se înscrie chiar și din confortul casei. De asemenea, cei care se confruntă cu probleme de sănătate ce necesită transfuzii constante pot depune o cerere după crearea unui cont în aplicație. Eficiența platformei provine din faptul că nu necesită administrare manuală, toate procesele desfășurându-se automat. Utilizatorii au control total asupra informațiilor lor medicale și personale, având posibilitatea de a face actualizări singuri. Astfel, aplicația noastră se dorește a fi un instrument simplu și accesibil, contribuind la îmbunătățirea procesului de donare și gestionare a sângelui pentru a salva mai multe vieți.



## 2. Analiza cerintelor utilizatorului:

Dorim să dezvoltăm un sistem de gestionare a donatorilor de sânge și a persoanelor care solicită sânge, inclusiv un istoric detaliat al acestora. Am identificat următoarele entități pentru a asigura acoperirea cât mai completă a informațiilor necesare donatorilor, beneficiarilor și pentru a eficientiza procesul de donare:

- Donatori și Primitori: pot să își creeze un user nou, asociat cu o parolă și un e-mail, precum și date personale, ca mai apoi să se logheze în baza de date cu informațiile aferente
- Info: stochează informațiile necesare unui donator sau unui primitor, precum: nume, prenume, cnp, sex, grupa sanguină
- Afecțiuni medicale primitori: conține o listă cu boli comune ce necesită transfuzii, precum și când a fost persoana în cauză diagnosticată și tratament
- Afecțiuni medicale donatori: cuprinde informații referitoare la afecțiunile medicale ale persoanelor care doresc să doneze sânge, inclusiv bolile existente sau cele dobândite ulterior, detaliile despre tratament și cantitatea maximă de sânge pe care aceștia o pot dona
- Cereri și Donații: se salvează un istoric al donațiilor de sânge aferente unui donator, precum și a cererilor pe care le depune un utilizator
- Centru: fiecare centru are o anumită cantitate de sânge asociată

În vederea atingerii obiectivelor propuse, am elaborat tabele distincte pentru a înregistra informații specifice despre fiecare donator în parte. Prin urmare, avem la dispoziție următoarele tabele, fiecare având structura detaliată prezentată în dreptul titlului:

- Centru: centruID, nume\_centru
- Stock: grupa\_sânge, cantitate, centruID
- Donator: donatorID, username, password, email
- Primitor: primitorID, username, password, email
- Info\_donator: donatorID, nume, prenume, cnp, gen, grupa\_sânge
- Info\_primitor: primitorID, nume, prenume, cnp, gen, grupa\_sânge
- Cereri: cerereID, primitorID, data\_cerere, cantitate, centruID
- Donații: donațieID, donatorID, data\_donație, cantitate, centruID
- Afecțiuni medicale: afecțiuneID, detalii
- Afecțiuni medicale donatori: donatorID, afecțiuneID, data\_diagnostic, tratament, cantitate\_maximă
- Afecțiuni medicale primitori: primitorID, afecțiuneID, data\_diagnostic, tratament



- Max\_donator: donatorID, cantitate\_donată

Platforma noastră implică două tipuri distincte de utilizatori, și anume donatori și primitori. Fiecare dintre acești utilizatori are statutul de utilizator intern și beneficiază de drepturi de acces la baza de date, obținute prin autentificarea în cont cu nume de utilizator și parolă.

- Donatorul:
  - Poate accesa și actualiza informațiile legate de starea lor de sănătate, istoricul donărilor și alte date relevante prin intermediul autentificării în cont
- Primitorul:
  - Poate crea solicitări pentru sânge, poate accesa informații despre disponibilitatea sângelui în anumite centre și poate actualiza informațiile privind starea sa de sănătate prin logarea în cont

### **3. Modelul de date si descrierea acestuia:**

#### **a. Tehnologii utilizate**

O bază de date reprezintă o modalitate de stocare a unor informații și date pe un suport extern (un dispozitiv de stocare), cu posibilitatea extinderii ușoare și a regăsirii rapide a acestora.

MySQL, o abreviere pentru „My Structured Query Language”, este un sistem de gestionare a bazelor de date relaționale (RDBMS) open-source, cunoscut pentru viteza, fiabilitatea și flexibilitatea sa. Acesta are ca scop inserarea datelor, interogații, actualizare și ștergere, modificarea și crearea schemelor, precum și controlul accesului la date.

O bază de date relațională, cum ar fi MySQL, organizează datele în tabele separate, în loc să le adună pe toate. Această abordare structurată asigură stocarea și recuperarea eficientă a datelor. Modelul logic de date, care cuprinde tabele de date, vizualizări, rânduri și coloane, oferă un mediu de programare flexibil. Regulile care definesc relațiile dintre câmpurile de date, cum ar fi unu-la-unu, unu-la-mulți, unic, obligatoriu sau opțional, împreună cu „indicatori” între diferite tabele, sunt stabilite pentru a menține consistența și integritatea datelor.

Operațiile de bază pe o bază de date sunt:

- crearea unui tabel

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE**  
**CATEDRA CALCULATOARE**

```
CREATE TABEL tbl_name ([data_type] [col_name], ...)
```

- afișarea datelor unui table

```
SELECT select_expr [, select_expr ...]
```

```
[FROM table_references [WHERE where_condition]
```

- inserarea în baza de date

```
INSERT [INTO] tbl_name [(col_name,...)] VALUE ({expr},...),(...),...)
```

- modificarea datelor

```
UPDATE table_reference
```

```
SET col_name1={expr1|DEFAULT} [, col_name2={expr2|DEFAULT}] ...
```

```
[WHERE where_condition]
```

- ștergerea datelor

```
DELETE FROM tbl_name
```

```
[WHERE where_condition]
```

În cadrul proiectului nostru, după cum am specificat și anterior, utilizăm următoarele tabele: `Afecțiuni_medicale`, `afecțiuni_medicale_donatori`, `afecțiuni_medicale_primitori`, `centru`, `cereri`, `donații`, `donator`, `info_donator`, `info_primitor`, `max_donator`, `primitor` și `stock`. Inițial, în codul nostru, am creat tabelele și am asignat cheie primară pentru fiecare, majoritatea fiind un ID, unic sau nu. Ulterior, am inserat date în acestea cu care vom lucra mai târziu.

Pentru a reuși să lucrăm cu baza noastră de date, am creat o serie de interogări, proceduri, `triggere` și `vederi`, fiecare în parte utilă pentru a rezulta un centru de donare sânge funcțional.

Interogări folosite:

- ✓ Selectarea informațiilor despre donatorii care au o anumită afecțiune de sânge
- ✓ Afișarea detaliilor despre cei care au donat într-un anumit interval de timp
- ✓ Afișarea numelui și adresei de email a primitorilor care au o anumită afecțiune medicală
- ✓ Afișarea cererilor de sânge efectuate de un anumit primitor
- ✓ Ordonarea donatorilor în funcție de data donării
- ✓ Găsirea totalului cantității de sânge donată de un anumit donator
- ✓ Afișarea informațiilor despre donatorii care nu au afecțiuni medicale
- ✓ Afișarea numelui și prenumelui primitorilor care au cerut o cantitate specifică de sânge de la un anumit centru
- ✓ Găsirea donatorilor care au donat în aceeași zi cu un alt donator în același centru
- ✓ Afișarea sumei stocurilor de sânge (per grupe) dintr-un anumit centru



**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE  
CATEDRA CALCULATOARE**

- ✓ Afișarea numărului total de donatori și primitori dintr-un anumit centru de donare
- ✓ Afișarea centrelor de donare care nu au cereri de sânge
- ✓ Afișarea celor care au donat cel mai mult sânge și cantitatea
- ✓ Afișarea donatorilor cu o anumită grupă de sânge care nu au afecțiuni medicale
- ✓ Afișarea cantității totale de sânge donată de fiecare grupă de sânge într-un anumit centru
- ✓ Afișarea celor care au primit sânge de la un anumit donator
- ✓ Afișarea detaliilor despre cei care au donat cel puțin o dată
- ✓ Afișarea donatorilor și primitorilor care au aceeași grupă de sânge
- ✓ Afișarea centrelor de donare care au cel mai mare stoc de sânge
- ✓ Afișarea numelui și adresei de email a celor care au donat de cel puțin 5 ori

Interogări scrise în algebră relațională:

- ✓ Afișarea numelui și prenumelui primitorilor care a solicitat o cantitate mai mare decât o anumită valoare
- ✓ Afișarea tuturor donatorilor care au aceeași grupă de sânge
- ✓ Afișarea tuturor cererilor de sânge care au fost efectuate într-un anumit interval
- ✓ Afișarea numărului total de donații efectuate de fiecare utilizator
- ✓ Afișarea detaliilor despre cei care au primit sânge într-un anumit interval

Proceduri folosite, în principal pentru interfață, pentru a avea un centru de donare sânge corespunzător:

- ✓ InregistrareDonator – pentru a adăuga un nou donator în listă
- ✓ InregistrareAfecțiuneDonator – pentru a adăuga o nouă afecțiune medicală în lista unui donator
- ✓ InregistrarePrimitor – pentru a adăuga un nou primitor în listă
- ✓ InregistrareAfecțiunePrimitor – pentru a adăuga o nouă afecțiune medicală în lista unui primitor
- ✓ InregistrareDonatie – pentru a adăuga în listă o nouă donație
- ✓ InregistrareCerere – pentru a adăuga în listă o nouă cerere
- ✓ RaportStocSânge – generează un raport al stocurilor de sânge dintr-un centru
- ✓ AfișareCentreSânge – afișează centrele care au disponibilă o anumită cantitate de sânge

Triggere utilizate:

- ✓ after\_donatie\_update\_stock - pentru actualizarea stocului după donație
- ✓ after\_cerere\_update\_stock - pentru actualizarea stocului după o cerere
- ✓ UpdateMaxDonator - actualizarea donatorului care a donat cea mai mare cantitate de sânge



**UNIVERSITATEA TEHNICĂ**

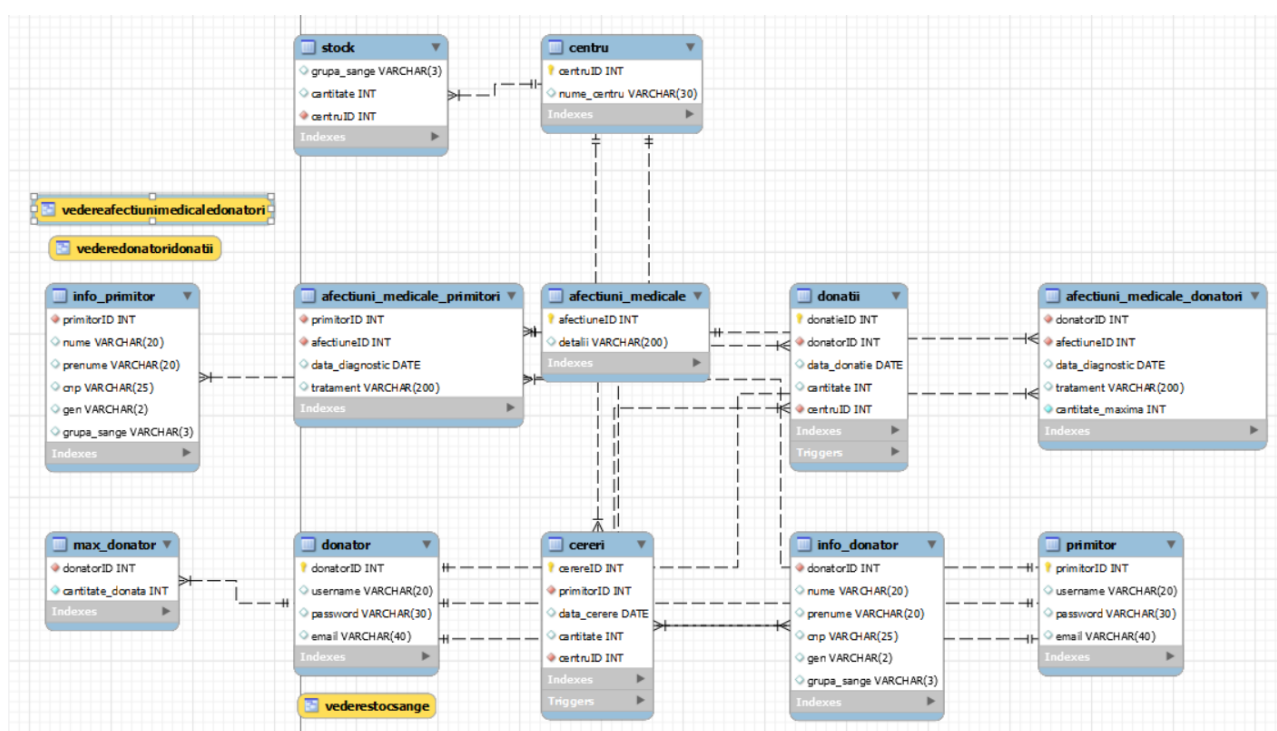
DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE**  
**CATEDRA CALCULATOARE**

Vederi folosite:

- ✓ VedereDonatoriDonatii – afișare informații despre donatori și donații
- ✓ VedereStocSange – afișare stoc curent de sânge
- ✓ VedereAfectiuniMedicaleDonatori – afișare afecțiuni medicale donatori

**b. Diagrama pentru modelul de date complet:**



Am presupus următoarele concepte pentru realizarea diagramei UML:

- Fiecare donator are o singură înregistrare asociată în info\_donator
- Fiecare primitor are o singură înregistrare asociată în info\_primitor
- Un donator sau un primitor poate avea mai multe afecțiuni
- O afecțiune poate fi asociată cu mai mulți utilizatori
- Un donator poate efectua mai multe donații
- O donație este realizată de un singur donator
- Un primitor poate depune mai multe cereri
- O cerere poate fi depusă de un singur primitor
- Un centru poate avea mai multe stocuri de sânge, pentru diferite grupe



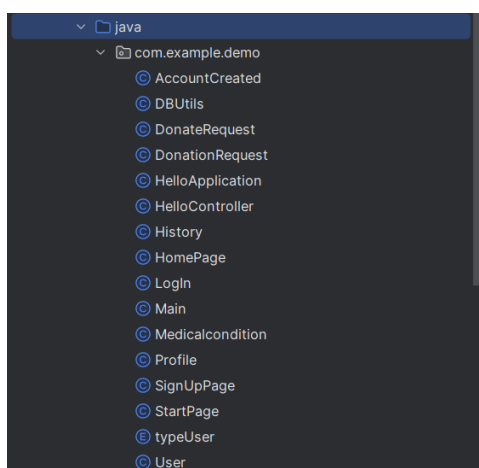


- Mai mulți donatori pot fi donatori\_max dacă au aceeași cantitate maximă de sânge donat

#### 4. Detalii de implementare:

Java este un limbaj de programare orientat-obiect, puternic tipizat, lansat în 1995. Cele mai multe aplicații distribuite sunt scrise în Java, iar noile evoluții tehnologice permit utilizarea sa și pe dispozitive mobile gen telefon, agenda electronică, palmtop etc. În felul acesta se creează o platformă unică, la nivelul programatorului, deasupra unui mediu eterogen extrem de diversificat. Acesta este utilizat în prezent cu succes și pentru programarea aplicațiilor destinate intranet-urilor. Limbajul împrumută o mare parte din sintaxă de la C și C++, dar are un model al obiectelor mai simplu și prezintă mai puține facilități de nivel jos. Un program Java compilat, corect scris, poate fi rulat fără modificări pe orice platformă care e instalată o mașină virtuală Java (Java Virtual Machine). Acest nivel de portabilitate (inexistent pentru limbaje mai vechi cum ar fi C) este posibil deoarece sursele Java sunt compilate într-un format standard numit cod de octeți care este intermediar între codul mașină (dependent de tipul calculatorului) și codul sursă.

JavaFX este o platformă software pentru crearea și furnizarea de aplicații desktop, precum și de aplicații web bogate care pot rula pe o mare varietate de dispozitive. JavaFX are suport pentru computere desktop și browsere web pe Microsoft Windows, Linux (inclusiv Raspberry Pi) și macOS, precum și dispozitive mobile care rulează iOS și Android, prin Gluon Mobile.



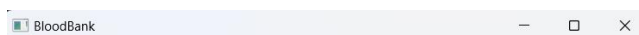
Clasele utilizate în acest proiect, cele legate de baza de date sunt DonateRequest, DonationRequest și MedicalCondition. Celelalte sunt folosite strict în realizarea interfeței, pentru a avea diferite funcționalități, cum ar fi pagina de start, pagina de sign-up, cea de log-in și ferestre separate pentru fiecare opțiune a aplicației.

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE**  
**CATEDRA CALCULATOARE**

## Interfața grafică:



Pagina de start, cea care se deschide inițial, unde putem alege dacă intrăm în aplicație ca donator sau ca primitor.

**DONOR**

**RECIPIENT**

Pe orice opțiune am intra, se deschide această fereastră de înregistrare în cont, unde introducem username-ul și parola, dacă avem cont. Dacă nu, alegem opțiunea **Create an account**, unde vom crea un cont nou în aplicație. În caz că datele cu care încercăm să ne logăm nu corespund, se va deschide o fereastră de eroare.

DonorLogin

[<Back](#)

**Log in into your account.**

**username**

**password**

**login**

Not a member? **Create an account!**

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE  
CATEDRA CALCULATOARE**

**Enter your details.**

First Name

Last Name

CNP

Gender

Blood Type

Username

email

Password

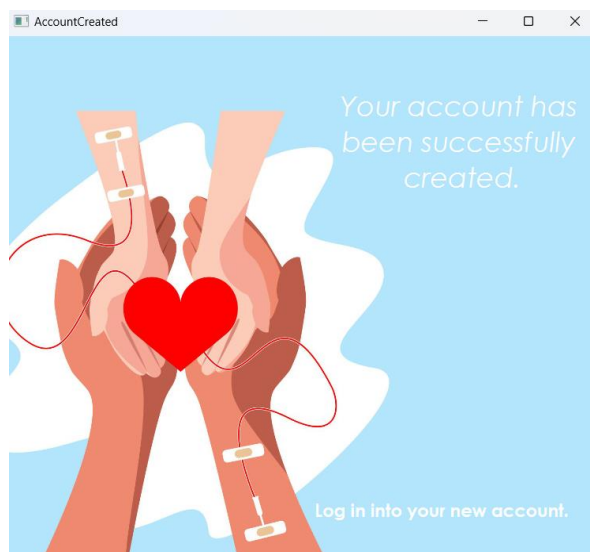
Confirm password

**Create your account.**

[<Back](#)

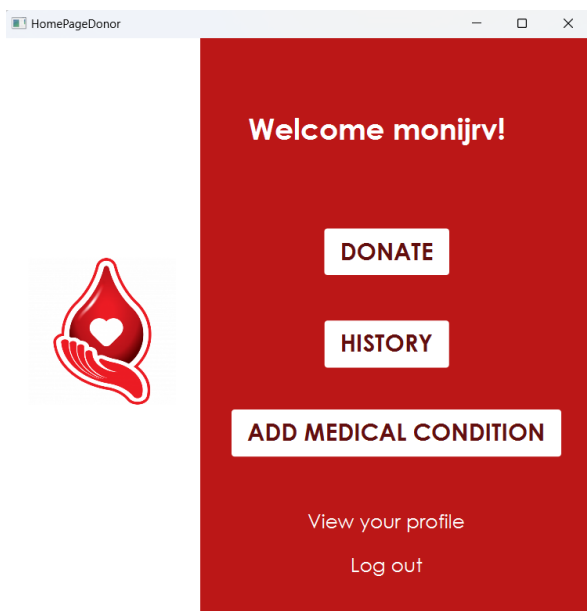
Această fereastră se deschide în cazul în care dorim să creăm un cont în aplicație, indiferent că este donator sau primitor. Aici vom completa fiecare câmp liber cu datele corespunzătoare și apăsăm butonul de **Create your account**. În caz că ne răzgândim, putem apăsa butonul **Back** și ne întoarce la fereastra anterioară.

Această pagină se deschide în cazul în care contul a fost creat cu succes, apoi apăsăm pe butonul din josul ferestrei pentru a ne întoarce la pagina de logare și a intra în noul cont.



**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

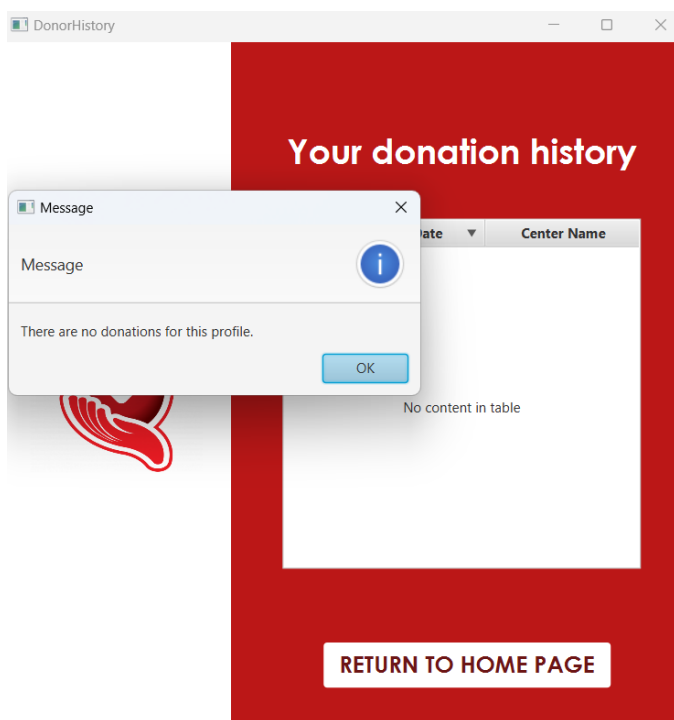
**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE**  
**CATEDRA CALCULATOARE**


Această fereastră se va deschide dacă logarea în cont a fost corectă. Aici, în funcție de butonul pe care îl selectăm, ne putem înscrie pentru o donație, putem vedea istoricul donațiilor, putem adăuga o nouă afecțiune medicală, putem vedea datele profilului nostru sau putem ieși din cont.

În această fereastră, putem înregistra online că vrem să facem o donație. Completăm câmpurile corespunzătoare, apoi selectăm opțiunea **Save**. În caz că ne răzgândim sau după ce salvăm cererea ne întoarcem la fereastra anterioară. În josul paginii apare cantitatea de sânge pe care o putem dona în funcție de afecțiunea medicală de care suferim.

**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

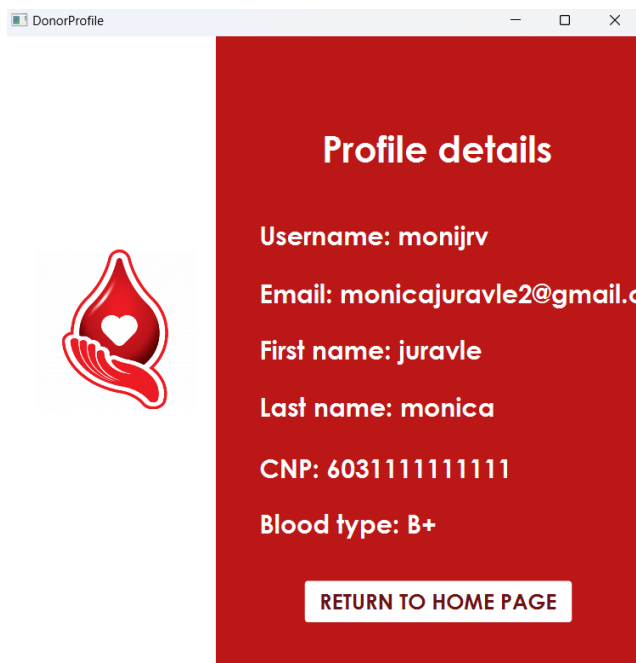
**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE  
CATEDRA CALCULATOARE**

Aici putem vedea istoricul donațiilor de până acum. În acest caz, apare o fereastră de mesaj pentru că nu există nici o donație asociată contului până în acest moment.

În această fereastră putem introduce o nouă afecțiune medicală în cazul că descoperim una. După ce completăm datele, apăsăm butonul de **Save** și ne întoarcem la fereastra anterioară.

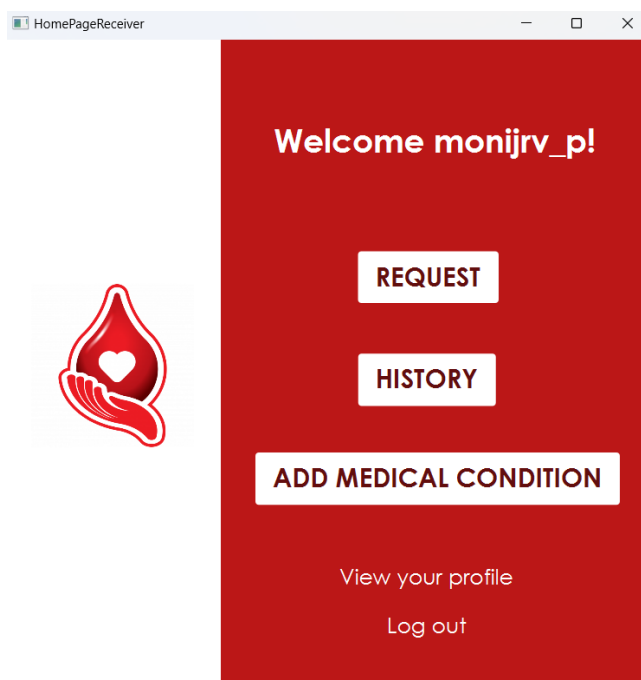
**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE**  
**CATEDRA CALCULATOARE**



Această fereastră se deschide în cazul că dorim să vedem datele contului în care suntem înregistrați. Pe butonul de jos ne întoarcem la fereastra anterioară.

Această fereastră este corespunzătoare contului de primitor, în cazul în care ne înregistrăm cu succes. De aici putem depune o cerere de sânge, putem vedea istoricul cererilor (similar celui de la donator) și de asemenea să adăugăm o nouă afecțiune medicală. În josul paginii, putem vedea detaliile contului în care ne aflăm (similar celui de donator) și putem ieși din cont.



**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE  
CATEDRA CALCULATOARE**

RequestBlood

### Blood request

Requested quantity

Date of request

Center

**SAVE**

**RETURN TO HOME PAGE**

Această fereastră este singura diferită față de cele pe care le putem deschide din contul de donator. Aici putem depune o cerere de sânge, după ce completăm câmpurile corespunzătoare și selectăm butonul **Save**.



## **5. Concluzii, limitări și dezvoltări ulterioare:**

Aplicația creată este într-o stare funcțională atât din punct de vedere al interfeței, cât și al bazei de date, dar pentru orice dezvoltări ulterioare, există riscul apariției unor probleme la nivelul interfeței în mare parte.

Posibilele dezvoltări ulterioare la nivelul bazei de date sunt adăugarea unor noi tabele cu cerințe suplimentare pentru donatori și primitori, cum ar fi: lista analizelor medicale anterioare, beneficiile primite în urma donării. De asemenea, putem adăuga un tabel de mesaje sau tabel cu personalul medical prezent în fiecare centru. Dezvoltări la nivelul interfeței se pot face conform tabelelor noi adăugate în baza de date.

La nivelul modului de acces în aplicație, se pot crea mai multe profiluri de utilizatori cum ar fi diferitele tipuri de staff ce lucrează într-un anumit centru. Tot la nivelul interfeței, se pot crea ferestre diferite după logare în funcție de nivelul de acces al utilizatorului.





**UNIVERSITATEA TEHNICĂ**

DIN CLUJ-NAPOCA

**FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE  
CATEDRA CALCULATOARE**

**6. Bibliografie/Resurse:**

- [http://ctsbucuresti.ro/dece\\_donez.html](http://ctsbucuresti.ro/dece_donez.html)
- [http://ro.wikipedia.org/wiki/Donare\\_de\\_s%C3%A2nge#Donarea\\_de\\_s.C3.A2nge\\_.C3.AEn\\_Rom.C3.A2nia](http://ro.wikipedia.org/wiki/Donare_de_s%C3%A2nge#Donarea_de_s.C3.A2nge_.C3.AEn_Rom.C3.A2nia)
- [http://ro.wikipedia.org/wiki/Baz%C4%83\\_de\\_date](http://ro.wikipedia.org/wiki/Baz%C4%83_de_date)
- <http://ro.wikipedia.org/wiki/SQL>
- <https://intuji.com/what-is-mysql-definition-comparision-with-sql-benefits-features-explained/>
- [http://ro.wikipedia.org/wiki/SQL#Elementele\\_limbajului\\_SQL](http://ro.wikipedia.org/wiki/SQL#Elementele_limbajului_SQL)
- [http://ro.wikipedia.org/wiki/SQL#Manipularea\\_datelor](http://ro.wikipedia.org/wiki/SQL#Manipularea_datelor)
- [http://ro.wikipedia.org/wiki/Java\\_\(limbaj\\_de\\_programare\)](http://ro.wikipedia.org/wiki/Java_(limbaj_de_programare))
- <https://en.wikipedia.org/wiki/JavaFX>