



UNIVERSITATEA TEHNICĂ "GHEORGHE ASACHI" IAȘI

FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

SPECIALIZAREA CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

- PROIECT LA DISCIPLINA BAZE DE DATE -

Sistem de gestiune și organizare al unei asociații de donare de sânge

Coordonator,

ș.l. dr.ing. Mironeanu Cătălin

Student,

Hrițcu Petronela-Adelina

grupa 1312B

Titlul proiectului

Sistem de gestiune și organizare al unei asociații de donare de sânge

Analiza, proiectarea și implementarea unei baze de date care să modeleze un sistem de gestiune al unei asociații ce se ocupă de donarea de sânge.

Descrierea proiectului – scopul aplicației

Donarea de sânge reprezintă un act altruist cu un impact direct asupra salvării vieților. Cu toate acestea, procesul de promovare, organizare și gestionare a campaniilor de donare poate fi complex, necesitând un nivel ridicat de coordonare și transparență. Proiectul „Asociație pentru donarea de sânge” are ca scop dezvoltarea unei platforme digitale inovatoare care să faciliteze accesul la informații și resurse pentru donatori și organizatori deopotrivă.

Prin intermediul acestei platforme, se dorește crearea unui mediu accesibil și sigur, care să sprijine fluxul constant de sânge către spitale și pacienți. Platforma promovează conștientizarea importanței donării de sânge, implicarea comunității și construirea unei rețele eficiente de solidaritate.

Obiectivele proiectului

1. **Facilitarea accesului la informații esențiale:** Donatorii și organizatorii vor avea acces rapid la date despre campaniile de donare active, locațiile disponibile și tipurile de donații acceptate.
2. **Simplificarea procesului de programare:** Crearea unei funcționalități care permite rezervarea locurilor la campaniile de donare în funcție de disponibilitate.
3. **Promovarea transparenței:** Oferirea de informații clare despre locațiile campaniilor, criteriile de eligibilitate și feedback-ul primit de la donatori.
4. **Consolidarea relației între donatori și organizatori:** Permițând crearea de conturi personale, unde donatorii pot urmări istoricul donațiilor și alte detalii relevante.
5. **Creșterea implicării comunității:** Prin campanii de conștientizare și facilitarea procesului de participare la campanii.

Structura și inter-relationarea tabelelor

În structura bazei de date dezvoltate au fost create următoarele entități:

Entitatea **donator** (ID_donator, nume, email, numar CI) conține informații generale despre donatori, având cheia primara id_donator (corelată cu tabelul detalii_donator).

1. id_donator (INTEGER, NOT NULL)

- Identificator unic pentru donator.
- INTEGER este ales pentru stocarea eficientă a valorii.

2. nume (VARCHAR2(100), NOT NULL)

- Numele complet al donatorului.
- Dimensiunea de 100 caractere este suficientă pentru nume complexe.

3. **email (VARCHAR2(100), NOT NULL)**

- Adresa de email a donatorului.
- Dimensiunea de 100 caractere este adecvată pentru adrese lungi.

4. **numar_ci (VARCHAR2(10), NOT NULL)**

- Numărul cărții de identitate.
- Dimensiunea de 10 caractere este suficientă pentru numerele CI.

Entitate **detalii_donator**(ID_donator, adresa, data_nasterii, greutate, inaltime, judet, oras, gen, telefon, serie, grupa_sanguina, data_ultimei_donatii) stochează date detaliate despre donatori, având cheia primară: id_donator.

1. **adresa (VARCHAR2(255), NOT NULL)**

- Adresa completă a donatorului.
- Dimensiunea de 255 caractere permite stocarea unei descrieri detaliate.

2. **data_nasterii (DATE, NOT NULL)**

- Data nașterii donatorului.
- Tipul DATE asigură validarea și manipularea corectă a datelor de tip dată.

3. **greutate (NUMBER(5, 2), NOT NULL)**

- Greutatea donatorului în kilograme.
- NUMBER(5, 2) permite valori de până la 999.99 kg, cu două zecimale, pentru precizie.

4. **inaltime (NUMBER(5, 2), NOT NULL)**

- Înălțimea donatorului în centimetri.
- NUMBER(5, 2) oferă suficient spațiu pentru valori precise (ex. 180.50 cm).

5. **judet (VARCHAR2(255), NOT NULL)**

- Județul de domiciliu al donatorului.
- Dimensiunea de 255 caractere permite flexibilitate în denumirile mai complexe.

6. **oras (VARCHAR2(255), NOT NULL)**

- Orașul de domiciliu al donatorului.
- Dimensiunea de 255 caractere este potrivită pentru diverse denumiri geografice.

7. **gen (CHAR(1), NOT NULL)**

- Genul donatorului: M (Masculin) sau F (Feminin).
- CHAR(1) este suficient pentru a stoca un singur caracter reprezentativ.

8. **telefon (VARCHAR2(15), NOT NULL)**

- Numărul de telefon al donatorului.
- Dimensiunea de 15 caractere este potrivită pentru numere internaționale.

9. **id_donator (INTEGER, NOT NULL)**

- Identificator unic al donatorului, referință către tabela donator.
- INTEGER asigură eficiența stocării.

10. **serie (CHAR(2), NOT NULL)**

- Serie specifică județului donatorului, validată pe baza regulilor definite.
- CHAR(2) este adecvat pentru a stoca doar două caractere.

11. **grupa_sanguina (CHAR(2), DEFAULT 'A', NOT NULL)**

- Grupa sanguină a donatorului (ex. A, AB, B, 0).
- CHAR(2) permite valori scurte și clare.

12. **data_ultimei_donatii (DATE)**

- Data ultimei donări efectuate de donator.

- Tipul DATE permite stocarea eficientă a informației.

Entitate **tip_donatie**(ID_tip_donatie, tip_donatie, descriere, criterii_eligibilitate) definește tipurile de donații, având cheia primară: id_tip_donatie.

1. **id_tip_donatie (INTEGER, NOT NULL)**
 - Identificator unic pentru tipul de donare.
2. **tip_donatie (VARCHAR2(50), NOT NULL)**
 - Denumirea tipului de donare.
3. **descriere (VARCHAR2(255))**
 - Descriere detaliată.
4. **criterii_eligibilitate (VARCHAR2(255))**
 - Criterii pentru eligibilitatea donării.

Entitate **campanie_donare**(ID_campanie, denumire_campanie, data, ID_locatie, numar_maxim_donatori) conține informații despre campaniile de donare, având cheia primară: id_campanie și cheia străină: id_locatie (legată de tabelul locatie).

1. **id_campanie (INTEGER, NOT NULL)**
 - Identificator unic pentru campania de donare. Este generat automat printr-o secvență, asigurând unicitatea.
 - Tipul INTEGER a fost ales pentru eficiență și scalabilitate.
2. **denumire_campanie (VARCHAR2(100), NOT NULL)**
 - Numele campaniei de donare.
 - Dimensiunea de 100 caractere permite includerea unor denumiri descriptive și detaliate.
3. **data (DATE, NOT NULL)**
 - Data desfășurării campaniei.
 - Tipul DATE asigură stocarea corectă și manipularea ușoară a valorilor de tip dată.
4. **numar_maxim_donatori (INTEGER, NOT NULL)**
 - Numărul maxim de donatori care pot participa la campanie.
 - Tipul INTEGER a fost ales pentru a reprezenta un număr întreg pozitiv.
5. **id_locatie (INTEGER, NOT NULL)**
 - Referință către locația în care are loc campania.
 - INTEGER este suficient pentru a stoca identificatorul unic al locației.

Entitate **programare**(ID_programare, ID_donator, ID_campanie, data_programare, status_programare, ID_tip_donatie)

1. **id_programare (INTEGER, NOT NULL)**
 - Identificator unic al programării.
2. **data_programare (DATE, NOT NULL)**
 - Data și ora programării.
3. **status_programare (VARCHAR2(20), DEFAULT 'Disponibila', NOT NULL)**
 - Statusul programării.
4. **id_tip_donatie (INTEGER, NOT NULL)**
 - Tipul donării (ex. sânge, plasmă).
5. **id_donator (INTEGER)**

- Referință către donatorul care a făcut programarea.

6. **id_campanie (INTEGER, NOT NULL)**

- Referință către campania asociată.

Entitate **feedback**(ID_feedback, ID_donator, ID_campanie, comentariu, nota)

1. **id_feedback (INTEGER, NOT NULL)**

- Identificator unic pentru feedback.
- INTEGER este eficient pentru stocarea unui identificator.

2. **comentariu (VARCHAR2(255))**

- Comentarii adiționale oferite de donator.
- Dimensiunea de 255 caractere oferă spațiu suficient pentru mesaje descriptive.

3. **nota (NUMBER(2, 1))**

- Nota acordată de donator, între 1.0 și 5.0.
- NUMBER(2, 1) permite o precizie de o zecimală.

4. **id_donator (INTEGER, NOT NULL)**

- Referință către donatorul care a acordat feedback-ul.

5. **id_campanie (INTEGER, NOT NULL)**

- Referință către campania evaluată.

Entitate **locatie**(ID_locatie, denumire, adresa, coordonate_geografice) conține informații despre locații, având cheia primară: id_locatie.

1. **id_locatie (INTEGER, NOT NULL)**

- Identificator unic al locației.
- Valoarea este generată automat printr-o secvență specifică.

2. **denumire (VARCHAR2(100), NOT NULL)**

- Denumirea locației.

3. **adresa (VARCHAR2(255), NOT NULL)**

- Adresa completă a locației.

4. **coordonate_geografice (VARCHAR2(50))**

- Coordonatele GPS (latitudine, longitudine).

Diagrama logică

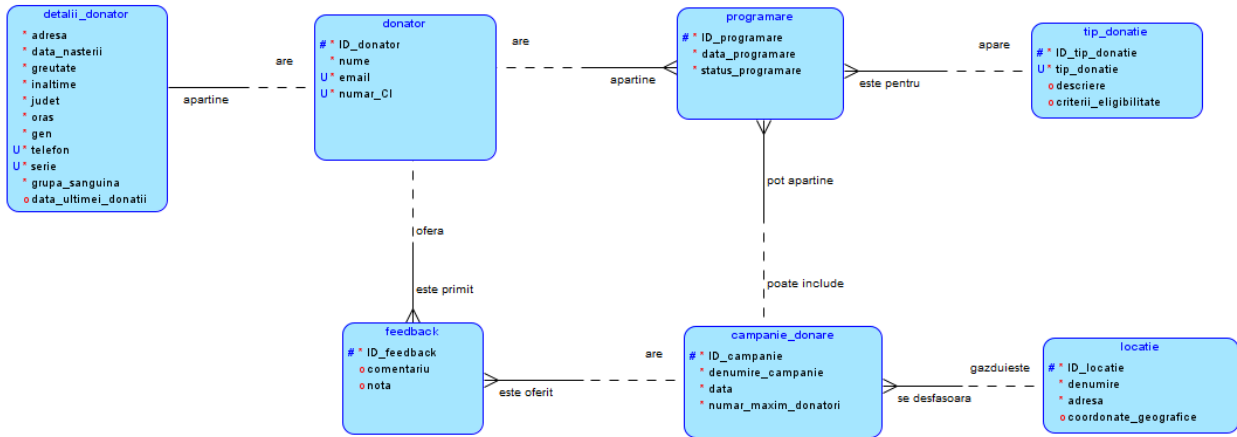
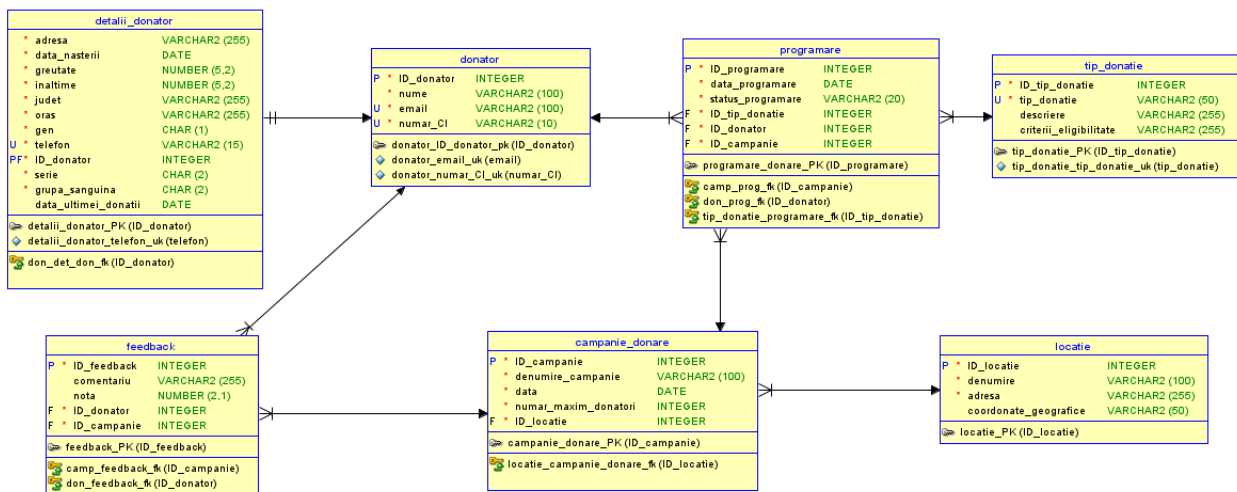


Diagrama relațională



Normalizare

Forma Normală 1

- **Tabelul detalii_donator:** câmpurile adresa, telefon, judet sunt valori atomice, nu colecții de valori, iar fiecare rând are o cheie primară (ID_donator), deci este unic.
- **Tabelul campanie_donare:** nu conține câmpuri repetitive (ex. nu există coloane pentru "Denumire 1", "Denumire 2"), iar fiecare campanie este identificată unic prin ID_campanie.

Toate tabelele sunt deja în 1NF, deoarece nu au attribute care să încalce principiile acestei forme normale.

Forma Normală 2

- **Tabelul detalii_donator:** toate coloanele (adresa, data_nasterii, greutate, etc.) sunt direct dependente de ID_donator. Nu există dependențe parțiale.
- **Tabelul programare:** cheia primară este ID_programare. Attributele precum data_programare, status_programare, ID_donator, ID_campanie depind complet de această cheie primară.

Modelul respectă 2NF, deoarece fiecare tabel conține attribute care depind complet de cheia primară.

Forma Normală 3

- **Tabelul campanie_donare:** attributele denumire_campanie, data, numar_maxim_donatori depind doar de ID_campanie, iar relația cu locația este gestionată prin ID_locatie, deci nu există dependențe tranzitive.
- **Tabelul detalii_donator:** adresa, greutate, data_nasterii, etc., depind doar de ID_donator, iar relația dintre serie și judet este verificată prin constrângeri, deci nu creează dependențe tranzitive.

Modelul este în 3NF, deoarece nu există dependențe tranzitive între attributele non-cheie.

În proiectul acestei baze de date există relații One-to-One și relații One-to-Many, după cum urmează:

Relația între tabele `detalii_donator` și `donator` este de tip 1:1. Fiecare donator are exact un set de detalii personale stocate în tabelul `detalii_donator`. Această relație asigură că informațiile detaliate sunt completate pentru fiecare donator în mod unic. Cheia straina este `detalii_donator.id_donator` → `donator.id_donator`.

Relația între `donator` și `feedback` este de tip 1:M. Fiecare donator poate oferi un singur feedback pentru o campanie specifică. Totuși, un donator poate participa la mai multe campanii, deci poate genera mai multe înregistrări în tabelul `feedback`, fiecare asociată unei campanii distincte. Cheia straina este `feedback.id_donator` → `donator.id_donator`.

Relația între `feedback` și `campanie_donare` este de tip 1:M. Fiecare campanie poate primi mai multe feedback-uri, fiecare asociat unui donator diferit. Fiecare feedback este asociat exact cu o campanie. Cheia straina este `feedback.id_campanie` → `campanie_donare.id_campanie`.

Relația între `donator` și `programare` este de tip 1:M. Un donator poate avea mai multe programări, dar fiecare programare este asociată cu o campanie specifică și un tip de donație unic. Cheia straina este `programare.id_donator` → `donator.id_donator`.

Relația între `campanie_donare` și `programare` este de tip 1:M. O campanie de donare poate avea mai multe programări asociate, fiecare corespunzând unui donator și unui tip de donație specific. Cheia straina este `programare.id_campanie` → `campanie_donare.id_campanie`.

Relația între `locatie` și `campanie_donare` este de tip 1:M. O locație poate găzdui mai multe campanii de donare, dar fiecare campanie se desfășoară într-o locație unică. Cheia straina este `campanie_donare.id_locatie` → `locatie.id_locatie`.

Relația între `tip_donatie` și `programare` este de tip 1:M. Fiecare tip de donație este asociat cu mai multe programări, dar o programare este asociată cu un singur tip de donație. Cheia straina este `programare.id_tip_donatie` → `tip_donatie.id_tip_donatie`.

Constrângeri

1. Constrângerea de cheie primară (PRIMARY KEY)

Fiecarui tabel îi este asignat un atribut cu o astfel de constrângere, deoarece asigură unicitatea fiecărei înregistrări din tabel și garantează că nu vor exista duplicări. De asemenea, acest lucru este esențial pentru a putea stabili relații între tabele folosind chei străine.

2. Constrângerea de cheie unică (UNIQUE)

- `donator.email`: constrângere unică pentru a asigura că nu există donatori cu același email
- `detalii_donator.telefon`: constrângere unică pentru a preveni duplicarea numărului de telefon
- `donator.numar_ci`: constrângerea asigură că fiecare număr de CI este unic pentru fiecare donator, astfel încât să se evite înregistrarea de multiple donatori cu același număr de CI

3. Constrângerea de tip (CHECK)

- `detalii_donator.grupa_sanguina`: trebuie să fie una dintre valorile ('O', 'A', 'AB', 'B') este necesară pentru a valida valoarea corectă a grupei sanguine, prevenind erorile de introducere a datelor și menținând integritatea informațiilor medicale esențiale
- `detalii_donator.greutate`: trebuie să fie între 50 și 120, deoarece greutatea donatorilor trebuie să fie într-un interval specific pentru a garanta că aceștia sunt eligibili pentru donare
- `programare.status_programare`: trebuie să fie unul dintre valorile ('Programata', 'Finalizata', 'Anulata', 'Disponibila'), deoarece este necesară pentru a controla statusul programării, prevenind erorile de înregistrare și asigurând claritatea și consistența datelor
- `detalii_donator.judet`: trebuie să fie un text care conține doar litere și eventuale spații între cuvinte (ex: 'Iași', 'București'), prevenind inserarea de date incorecte în câmpul județului, asigurându-se că numele județului este introdus corect și respectă formatul corect de litere și spații.
- `detalii_donator.oras`: trebuie să fie un text care conține doar litere și eventuale spații între cuvinte (ex: 'Cluj-Napoca', 'Timișoara').
- `detalii_donator.gen`: trebuie să fie una dintre valorile ('M', 'F'), pentru a garanta că valoarea introdusă pentru sex este validă
- `detalii_donator.telefon`: trebuie să fie un număr de telefon format din exact 10 cifre, deoarece asigură că numărul de telefon este valid și că respectă formatul standard (exact 10 cifre), ceea ce este important pentru contactarea corectă a donatorilor
- `detalii_donator.serie`: trebuie să aibă exact 2 caractere și să respecte regulile specifice județului (de exemplu, pentru județul Iași sunt permise doar seriile 'MX' și 'MZ'). Această constrângere validează seria de buletin/CI, având în vedere regulile specifice fiecărui județ, asigurându-se că datele sunt conforme cu legislația locală.
- `campanie_donare.numar_maxim_donatori`: trebuie să fie mai mare decât 0. Constrângerea impune ca numărul maxim de donatori pentru o campanie să fie mai mare decât 0, astfel încât să se evite campaniile fără limită de donatori sau cu o limită incorectă.
- `feedback.nota`: trebuie să fie între 1.0 și 5.0. Constrângerea asigură că notele acordate feedback-urilor se află în intervalul corespunzător (1.0-5.0), astfel încât să se evite notele nerealiste sau eronate, asigurându-se astfel că evaluările sunt în limitele acceptate.

Triggere pentru autoincrementare

Triggerele de autoincrement din cod atribuie automat valori unice pentru ID-urile primare ale tabelelor utilizând secvențe definite anterior. Acestea sunt declanșate înaintea operațiilor de inserare și asigură că fiecare înregistrare primește un ID incremental fără intervenție manuală.

Triggerele de autoincrement folosite în cod sunt:

1. **campanie_donare_id_campanie** – pentru atribuirea unui ID unic în tabela `campanie_donare`.
2. **donator_id_donator_trg** – pentru atribuirea unui ID unic în tabela `donator`.
3. **feedback_id_feedback_trg** – pentru atribuirea unui ID unic în tabela `feedback`.
4. **locatie_id_locatie_trg** – pentru atribuirea unui ID unic în tabela `locatie`.
5. **programare_id_programare_trg** – pentru atribuirea unui ID unic în tabela `programare`.
6. **tip_donatie_id_tip_donatie_trg** – pentru atribuirea unui ID unic în tabela `tip_donatie`.

Triggerele pentru validarea datelor

1. `Trg_campanie_donare`:
 - Scop: Verifică dacă data campaniei (`data`) este mai mare decât data curentă (`SYSDATE`). Dacă data este mai mică sau egală cu data curentă, se aruncă o eroare.
 - Utilitate: Asigură că toate campaniile de donare sunt programate pentru viitor, prevenind inserarea unor campanii cu dată invalidă.
2. `Trg_data_programare`:
 - Scop: Verifică data programării (`data_programare`) în funcție de starea programării (`status_programare`):
 - Dacă `status_programare` este 'Disponibila', data trebuie să fie în viitor.
 - Dacă `status_programare` este 'Finalizata', nu se face nicio verificare suplimentară pentru dată.
 - Utilitate: Asigură că programările cu statusul „Disponibilă” au o dată validă în viitor.
3. `Trg_detalii_donator`:
 - Scop: Verifică dacă data nașterii (`data_nasterii`) a donatorului este mai mică decât data curentă (`SYSDATE`). Dacă nu, se aruncă o eroare.
 - Utilitate: Previne inserarea de date de naștere viitoare, asigurându-se că doar date valide sunt acceptate pentru donatori.

Descriere use-case-uri

1. **Crearea unui cont de donator** - permite unui utilizator să creeze un cont de donator în sistem, înregistrându-se cu datele personale (nume, email, număr de identitate) și detalii medicale (grupă sanguină, greutate, înălțime etc.). Tranzacția verifică validitatea datelor și asigură că informațiile sunt corecte și conforme cu regulile de validare.
2. **Programarea unei donații** - acest usecase are scopul de a crea o programare pentru un donator, respectând mai multe condiții de validare înainte de a înregistra efectiv

programarea. Iată pașii verificați și logica utilizată pentru a asigura că donatorul îndeplinește toate cerințele necesare:

- Verificarea existenței donatorului în sistem: se verifică dacă donatorul este înregistrat în sistem printr-o interogare în tabelul donator, pe baza ID-ului donatorului; dacă donatorul nu este găsit în sistem, se aruncă o eroare cu mesajul "Donatorul nu este înregistrat în sistem!".
- Verificarea eligibilității donatorului pentru o nouă donație (în funcție de ultima donație): se verifică dacă donatorul a făcut o donație în ultimele 3 luni. Această verificare se face prin accesarea câmpului data_ultimei_donatii din tabelul detalii_donator; dacă ultima donație a fost făcută mai recent de 3 luni, se aruncă o eroare cu mesajul "Donatorul nu poate face o nouă donație decât la 3 luni după ultima donație!".
- Verificarea disponibilității locurilor pentru campania respectivă: se calculează numărul de locuri disponibile pentru o anumită campanie și dată. Acest lucru se face printr-o interogare care determină câte locuri s-au ocupat deja și le scade din numărul maxim de locuri pentru campanie; dacă nu sunt locuri disponibile, se aruncă o eroare cu mesajul "Nu mai sunt locuri disponibile pentru această dată!".
- Verificarea existenței unei programări anterioare la aceeași dată: se verifică dacă donatorul are deja o programare pentru aceleași date în tabelul programare.
- Dacă toate verificările anterioare sunt trecute cu succes, se creează o nouă intrare în tabelul programare, asociind donatorul cu campania respectivă și stabilind statusul ca "Programata".
- După înregistrarea programării, numărul de locuri disponibile pentru campania respectivă este actualizat, scăzându-se un loc din câte locuri erau disponibile anterior.

3. **Lăsarea unui feedback pentru o campanie** - permite unui donator să lase un feedback pentru o campanie la care a participat. Feedback-ul este validat pentru a se asigura că donatorul a participat efectiv și că feedback-ul este lăsat într-un termen corect (în ultimele 30 de zile după donație).
4. **Permiterea vizualizării de către utilizatorii platformei de donare de sânge a feedback-urilor disponibile pentru toate campaniile de donare** - afișează toate feedback-urile lăsate de donatori pentru toate campaniile, împreună cu informațiile suplimentare despre donatori și campanii.
5. **Anularea unei programări** - un donator decide să își anuleze o programare înainte de data stabilită, statusul programării este schimbat din "Programată" în "Anulată".
6. **Ștergerea unui cont de donator** - un donator decide să își șteargă contul din platformă.
7. **Actualizarea statusului unei programări** - după finalizarea donației, statusul programării este schimbat din "Programată" în "Finalizată".

8. **Actualizare data ultimei donatii unui donator** – daca un donator are o programare cu statusul „Finalizata”, se va actualiza data ultimei donatii a acestuia cu data ultimei programari finalizate.
9. **Permiterea vizualizării de către toți utilizatorii platformei a campaniilor disponibile, a locatiilor disponibile de donare, dar si a numarului maxim de donatori suportat.**