**ACADEMIA DE STUDII ECONOMICE DIN BUCUREŞTI**

**FACULTATEA DE CIBERNETICĂ, STATISTICĂ ŞI INFORMATICĂ ECONOMICĂ**

**LUCRARE DE LICENȚĂ**

**INTEGRAREA TEHNOLOGIILOR WEB CU BAZE DE DATE ORACLE ÎNTR-O APLICAȚIE ANDROID**

**Coordonator ştiinţific:**

**Conf. Univ. Dr. IULIANA BOTHA (ŞIMONCA)**

**Absolvent:**

**LEONTE ALEXANDRU**

**BUCUREŞTI**

**2019**

**CUPRINS**

[INTRODUCERE 4](#_Toc1070826455)

[CAPITOLUL 1. ANALIZA SISTEMULUI DE TRANSFUZIE SANGUINĂ ÎN ROMÂNIA 5](#_Toc1720197954)

[1.1 Noțiuni fundamentale cu privire la domeniul hematologiei (ramura transfuziei sanguine) 5](#_Toc151912347)

[1.1.1 Sistemul AB0 6](#_Toc1891769675)

[1.1.2 Sistemul Rh 7](#_Toc2056787652)

[1.2 Importanța utilizării unui sistem informatic în cadrul circuitului transfuziei sanguine 8](#_Toc1532988957)

[1.3 Analiza sistemelor informatice existente 9](#_Toc870622412)

[CAPITOLUL 2. ANALIZA CERINȚELOR SISTEMULUI INFORMATIC 12](#_Toc99246796)

[2.1 Cerințe funcționale ale unui sistem informatic în domeniul transfuziei sanguine 12](#_Toc919866464)

[2.2 Diagrame UML 13](#_Toc448362383)

[2.2.1 Diagrame ale cazurilor de utilizare 13](#_Toc779172129)

[2.2.2 Diagrame de activitate 18](#_Toc1444484200)

[2.2.3 Diagrama de clase 19](#_Toc1760370573)

[2.2.4 Diagrame de stări 20](#_Toc786909149)

[CAPITOLUL 3. PROIECTAREA SISTEMULUI INFORMATIC 21](#_Toc734738513)

[3.1 Rafinarea diagramelor UML 21](#_Toc1172066051)

[3.2 Proiectarea bazei de date 22](#_Toc1441945287)

[3.3 Proiectarea ierarhiei de ferestre 26](#_Toc884106934)

[3.4 Proiectarea interfețelor utilizator 26](#_Toc577090932)

[3.5 Diagrama de componente 28](#_Toc1211055367)

[3.6 Diagrama de desfășurare 28](#_Toc307236837)

[3.7 Tehnologiile informatice utilizate 29](#_Toc1646319363)

[3.7.1 Limbajul SQL 29](#_Toc697039939)

[3.7.2 Serviciile REST 30](#_Toc1693688104)

[3.7.3 Formatele de reprezentare 30](#_Toc1352315925)

[3.7.4 Limbajul Java 31](#_Toc1764907154)

[3.7.5 Tehnologiile Servlet și JSP 32](#_Toc422666812)

[3.8 Produse informatice utilizate 32](#_Toc1139108136)

[3.8.1 Oracle VM VirtualBox 33](#_Toc1585155675)

[3.8.2 SQL Developer 33](#_Toc231559650)

[3.8.3 NetBeans 33](#_Toc1481668123)

[3.8.4 Android Studio 33](#_Toc508498482)

[CAPITOLUL 4. IMPLEMENTAREA ȘI PREZENTAREA SISTEMULUI INFORMATIC 34](#_Toc1951757604)

[Bibliografie 55](#_Toc1633580471)

[Anexe 58](#_Toc252784509)

[Anexa 1. Indexul termenilor utilizaţi 58](#_Toc1861061608)

[Anexa 2. Indexul figurilor 60](#_Toc1019085780)

# INTRODUCERE

Având în vedere situația curentă în ceea ce privește sistemul de transfuzie sanguină în România, precum și nivelul de cunoaștere a subiectului de către cetățeni, se impune implementarea unui ansamblu de produse informatice care, pe de o parte să faciliteze procesul din perspectiva centrelor de colectare sanguină, iar pe de altă parte să contribuie la conștientizarea populației cu privire la problemele ridicate de lipsa volumului necesar de sânge în circuitul sistemului sănătății.

În acest sens, lucrarea va acoperi toate aceste problematici, în încercarea de a găsi o modalitate eficientă de a îmbunatăți sistemul curent. Astfel, vor exista două abordări fundamental diferinte, una dinte ele urmând să trateze gestiunea centrelor de transfuzie sanguină, în timp ce a doua va pune accentul pe modul de informare și de agregare a potențialilor donatori.

Astfel, acest studiu de caz își are rădăcinile în domeniul medical, produsele informatice implementate urmărind să rezolve o problematică socială în contextul perfecționării uneia dintre cele mai importante ramuri angrenate în funcționarea cât mai eficientă a societății.

Această lucrare se va axa pe ciclul de dezvoltare al unui sistem informatic, urmând a fi structurată în patru capitole, ilustrându-se partea de identificare a cerințelor, partea de analiză, partea de proiectare și partea de implementare.

În cadrul primului capitol vor fi expuse aspectele fundamentale ale domeniului transfuziei sanguine, precum și motivele pentru care este necesară implementarea unui astfel de produs informatic în România.

Cel de-al doilea capitol va prezenta procesul de proiectare a sistemului informatic, diagramele UML mapându-se pe cerințele funcționale identificate anterior.

Următorul capitol prezintă aspectele necesare proiectării ansamblului informatic, precum proiectarea interfețelor de utilizator și a modului în care acestea vor interacționa între ele, fiind detaliate produsele și tehnologiile informatice care vor fi utilizate în viitor.

Ultimul capitol va conține atât secvențe de cod, cât și capturi de ecran realizate aplicațiilor finale, împreună cu explicațiile funcționalităților.

# CAPITOLUL 1. ANALIZA SISTEMULUI DE TRANSFUZIE SANGUINĂ ÎN ROMÂNIA

## 1.1 Noțiuni fundamentale cu privire la domeniul hematologiei (ramura transfuziei sanguine)

Hematologia, din punct de vedere etimologic (conform [POAN05]), provine din greaca antică, din alipirea cuvântului “haima”, care de-a lungul timpului a căpătat forma “hemato” (sânge), și a cuvântului “logos” (știință).

Astfel, se poate deduce definiția hematologiei, ea fiind o ramură a biologiei care se ocupă cu studiul în ansamblu al tuturor aspectelor posibile ale sângelui, fie că vorbim de componentele acestuia, de funcțiile pe care le îndeplinește în cadrul aparatului cardiovascular sau de bolile care pot decurge din acesta.

Se poate afirma că hematologia este un domeniu extrem de complex, el fiind de altfel împărțit intr-o multitudine de alte ramuri, printre care și cel al transfuziei sanguine.

Transfuzia sanguină este o ramură a biologiei interne care se ocupa cu studiul gestiunii transferului de sânge și cu analiza componentelor acestuia, de-a lungul timpului ea reușind să descopere regulile care trebuie urmărite pentru a evita apariția numeroaselor complicații în momentul realizării transfuziilor, proces care de altfel este unul extrem de obișnuit în viața de zi cu zi.

Cel mai important aspect care apare în cadrul acestui domeniu este cel al grupelor sanguine. Conform [NET01], Societatea Internaționala a Transfuziei Sanguine (ISBT) recunoaște în momentul de față 36 de tipuri de clasificare a grupelor sanguine în funcție de antigene, numarul acestora fiind de 360. Aceasta informație oficială confirma afirmația întalnită în cadrul [NOSM13], autorul susținând existența a peste 300 de grupe sanguine, dar și faptul ca foarte puține dintre acestea diferă fundamental de restul, ajungându-se foarte rar în situații în care acestea provoacă reacții în urma transfuziilor.

Având în vedere acest aspect, de-a lungul timpului s-a încercat acoperirea cât mai mare a grupelor fundamentale prin utilizarea a cât mai puține tipuri de clasificări, în ziua de azi cele mai importante sisteme de clasificare fiind AB0 si Rh.

### Sistemul AB0

Sistemul AB0 urmărește prezența a două aglutinogene (antigeni) la nivelul eritrocitelor (globulelor roșii/hematii), dar și a existenței a două aglutinine (anticorpi), care sunt produse de către plasma sanguină în momentul în care individul este expus la o cantitate de sânge care aparține de altă grupă față de cea proprie (în cazul transfuziilor sau nașterilor).

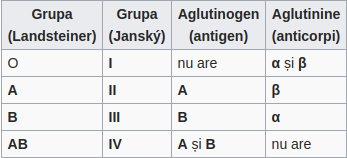


Figura 1.1 Componentele pe fiecare grupă sanguină (Wikipedia)

Figura 1.1 expune distribuția acestor elemente în cadrul fiecărei grupe de sânge principale identificate de sistemul AB0: 0(I), A(II), B(III) și AB(IV).

Regula de bază care definește acest sistem este aceea că antigenele existente în globulele roșii ale unei persoane nu pot coexista cu anticorpii omologi prezenți în plasmă (A - alfa, B - beta). În cazul încălcării acestei reguli, individul poate suferi de hemoliză (distrugerea eritrocitelor). (adaptat din [NOSM13])

Genetica acestui sistem presupune existența a trei tipuri de gene, două dintre ele fiind codominante (IA si IB), iar ultima fiind una recesiva (I0 sau i). Grupele de sânge conțin doua astfel de gene și se determină în funcție de genotipul reprezentat de combinația acestora (Figura 1.2). Cu ajutorul acestei reprezentări se poate stabili și grupa de sânge a descendenților, modul de funcționare permițănd păstrarea unei anumite diversități în structura în populație a distribuției grupelor sanguine. (adaptat din [NET02]).

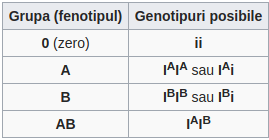


Figura 1.2 Reprezentarea genotipurilor pe grupă sanguină (Wikipedia)

Totuși, sistemul AB0 nu este suficient de unul singur pentru a putea explica, analiza și trata problematica cât mai eficient, acestui sistem alăturându-i-se astfel sistemul Rh.

### Sistemul Rh

Sistemul Rh urmărește prezenta factorului Rh (Rhesus), identificat cel mai des de către antigena D din cadrul hematiilor din pricina ușurinței cu care poate fi depistată. Astfel, indivizii care posedă acest factor sunt plasați în cadrul Rh+, în timp ce restul aparțin de Rh-.

Indivizii care au un Rh negativ generează anticorpi anti-D în momentul în care se realizează o transfuzie cu sânge cu Rh pozitiv, lucru care ar duce la producerea hemolizei, prin prezența genelor omoloage D - anti-D. (adaptat din [NOSM13])

Astfel, prin coroborarea regulilor celor doua sisteme prezentate, transfuziile sanguine urmăresc un set de reguli care indică toate modurile posibile în care pot fi realizate (Figura 1.3).

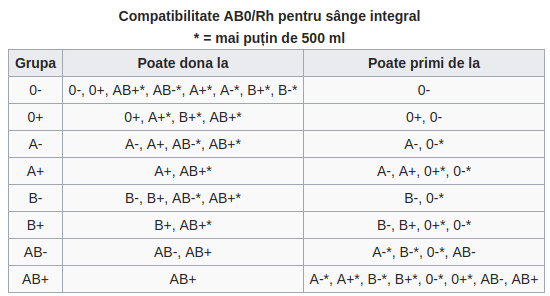


Figura 1.3 Compatibilitățile grupelor (Wikipedia)

Potrivit [DEAN05], constrângerea de <500ml este generată de prezența în plasma sanguină a anticorpilor, care în aceste cantități este îndeajuns de diluată pentru a nu produce hemoliza în momentul în care intră în contact cu aglutinogenele omoloage .

De asemenea, există posibilitatea separării celor două componente sanguine, a plasmei și a hematiilor, care poate aduce anumite avantaje în cazul îndeplinirii unor condiții stricte.

## Importanța utilizării unui sistem informatic în cadrul circuitului transfuziei sanguine

Având în vedere toate aspectele teoretice prezentate în cadrul subcapitolului 1.1, se poate deduce faptul că, pentru a putea satisface întregul mecanism al transfuziei sanguine, este nevoie de o metodă de informare cât mai eficientă și un canal de distribuție cât mai adecvat societății contemporane.

Acest lucru se datorează restricțiilor cauzate de regulile de transfuzie specifice celor două sisteme prezentate, anumite grupe având o frecvență de apariție extrem de scăzută, acest lucru conducând implicit la existența unei rezerve mult mai mici de volum de sânge disponibila pentru anumiți indivizi.

De exemplu, potrivit [CRSA14], sistemul Rh are la nivel mondial o structură de aproximativ 85% reprezentată de Rh+, în timp ce Rh-ul negativ acoperă doar 15% din populație. Ținând cont de restricțiile impuse prin sistemul Rh, cei care nu prezintă pe hematii antigenele specifice ar putea primi prin transfuzie masă sanguină de la cel mult 15% de oameni, acest procentaj fiind ulterior influențat și de ramificarea pe sistemul AB0.

La nivelul României, situația se prezintă conform Figurii 1.4:





Figura 1.4 Structura grupelor sanguine în România ([NET03])

Se poate observa urmărirea trendului procentajelor prezentate mai sus în cadrul sistemului Rh, în România situația gravitând în jurul procentajelor de 86-14.

Dacă grupele care prezintă un Rh pozitiv nu întâmpină neaparat probleme când vine vorba de posibili donatori (chiar dacă grupa AB+ are o distribuție de “doar” 7%, ea este adevăratul receiver universal, întrucât poate primi sânge de la orice grupă), nu același lucru se poate specifica și despre cele cu Rh negativ (deși conform sistemului AB0, grupa AB este receiver universal, acest concept poate fi interpretat greșit, grupa AB- având ca potențiali donatori doar 14% din populație, în timp ce grupa 0- este extrem de vulnerabilă, fiind acoperită de doar 5%).

În acest sens vin nenumăratele apeluri către cetățeni din partea persoanelor din sistemul medical, potrivit ministrului sănătății, citat de [NET04], la data de 24 iulie 2018 în București se prezentau în medie 120 de persoane la centrul de transfuzie, în timp ce estimarea de donatori necesari era în jurul valorii de 500.

De asemenea, potrivit [NET05], România nu strălucește în ceea ce privește procentajul persoanelor adulte care optează sa doneze, doar 1.7% dintre aceștia prezentându-se la centrele de transfuzie.

Cauzele principale au fost determinate de către [NET06], printre acestea numărăndu-se cunoașterea precară a subiectului de către cetățeni, precum și indisponibilitatea acestora, datorată lipsei timpului.

Chiar dacă se caută implementarea unor soluții care să trateze toate aceste probleme, anumite exemple fiind mobilizarea prin campaniile mobile, prin expunerea mesajelor informative în spațiul public, situația nu pare să se îmbunătățească.

Potrivit dnei Doina Gosa, Directorul Centrului de Transfuzii din București, citată în cadrul [NET06], principala problemă a sistemului curent este numărul scăzut de astfel de centre, dar și a personalului redus de care se dispune. Astfel, cantitatea de sânge colectată este una care nu satisface cererea spitalelor, foarte des ajungându-se în situația în care sunt lansate în spațiul public alerte de către spitale sau chiar de către persoanele apropiate celor care necesită sânge.

Astfel, ansamblul de produse informatice care va fi produs în cadrul acestei lucrări va urmări să aducă un plus sistemului curent, fiind tratate ambele capete ale problemei.

Pe de o parte, se va încerca îmbunătățirea procesului urmat de centrele de tranfuzie, fie că vorbim de gestionarea cât mai eficientă a resurselor de care dispun sau de informarea publică din punctul de vedere al necesității de cantitate de sânge.

Pe de altă parte, va fi tratată problematica din perspectiva potențialilor donatori, urmând a fi implementate module de informare în ceea ce privește situația actuală, dar și pașii și procedurile care trebuie urmate pentru a fluidiza circuitul transfuziei sanguine.

## Analiza sistemelor informatice existente

În urma analizării produselor informatice care tratează această problematică s-a putut observa faptul că marea majoritate a acestora abordează subiectul doar din punctul de vedere al informării și atragerii potențialilor donatori.

În plus, un alt aspect identificat a fost acela că în România, la momentul actual există un număr infim de astfel de solutii. Având în vedere faptul că cea mai completă dintre acestea (NB - aplicația Donorium) a fost lansată în mediul public la momentul redactării acestei lucrări, dar și modul de realizare a acesteia (în coordonare cu centrele de transfuzie sanguină), putem afirma faptul că nevoia de tehnologizare a acestui mecanism era una absolut necesară și în același timp, iminentă.

Doina Gosa, Directorul Centrului de Transfuzii din București, citată de [NET07], declară: “*Mă bucură foarte mult pentru interesul [...] manifestat pentru aplicație* (NB - aplicația Donorium). *Îi așteptăm [...] să se programeze și să doneze. Ne dorim foarte mult să putem să lucrăm pe viitor doar în baza programărilor la donare. Acest lucru este benefic și pentru donatori, care vor petrece mai puțin timp să doneze, și pentru doctori, care vor putea avea un flux constant de consultații.*”.

Această afirmație nu poate decât să întărească motivele sugerate în cadrul capitolului 1.2 (Importanța utilizării unui sistem informatic în cadrul circuitului transfuziei sanguine), fiind invocată îmbunătățirea procesului atât din perspectiva donatorilor, dar și din cea a personalului medical.

Din punctul de vedere al funcționalităților oferite, majoritatea aplicațiilor oferă un mediu de informare în ceea ce privește condițiile de îndeplinit pentru a putea fi eligibil pentru donare, dar și o listă a centrelor de transfuzie disponibile, precum și modalități de intrare în contact cu acestea. Cu toate acestea, majoritatea nu oferă un canal de distribuție a posibilelor alerte lansate de centrele de transfuzie, aspect esențial de tratat pentru a putea exploata o astfel de implementare informatică la cel mai înalt nivel (NB - doar Donorium + [Sanguinity](https://start-up.ro/proiecte-romanesti-pentru-startup-uri-in-sanatate/" \t "/home/alexsalupa97/Documents\\x/_blank)).

De asemenea, se impune existența unei soluții informatice care să îmbunătățească procesul desfășurat de centrele de transfuzie, în speță stocarea cantităților de sânge disponibile și gestionarea cât mai eficientă a resurselor existente. La momentul actual există un singur astfel de produs, “CTS Manager”, acesta fiind menționat în cadrul [NET08], neexistând însă o descriere detaliată care să expună toate funcționalitățile oferite. Astfel, efectuarea unei analize comparative pertinente nu este posibilă. Mai mult, este necesară existența unui feature care să permită generarea rapoartelor cu privire la fiecare centru de transfuzie în parte, pentru a putea avea o imagine de ansamblu cât mai clară la fiecare moment de timp.

Ansamblul de produse informatice dezvoltate în cadrul acestei lucrări își propune să ofere un set de funcționalități indispensabil procesului actual, fiind urmărită implementarea acestora într-un mod logic, iar modul de interacțiune cu acestea să fie unul fluid. În acest fel, rezultatul final va putea avea un impact benefic din punct de vedere social.

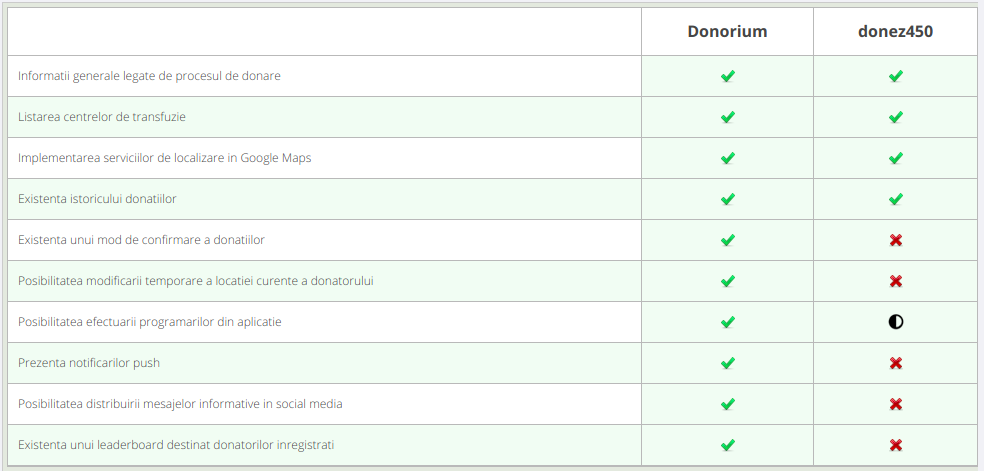


Figura 1.5 Tabel comparativ

# CAPITOLUL 2. ANALIZA CERINȚELOR SISTEMULUI INFORMATIC

## **2.1 Cerințe funcționale ale unui sistem informatic în domeniul transfuziei sanguine**

Ca urmare a identificării și analizării cerințelor domeniului transfuziilor sanguine, următorul pas al procesului a fost acela de a dezvolta o serie de cerințe funcționale care să trateze adecvat problematica.

Astfel, am identificat 3 posibile tipuri de utilizatori, după cum urmează: centre de transfuzie, donatori și pacienți.

Centrele de transfuzie vor declara la crearea unui cont volumul disponibil pe fiecare grupă de sânge de care dispun la momentul respectiv, precum și limitele inferioare ale acestora.

În cazul în care aceste limite vor fi atinse, se vor declanșa alerte către donatorii din orașul respectiv care dețin grupe de sânge compatibile pentru donare către grupa cu probleme, situațiile curente ale centrelor de transfuzie urmând a fi verificate zilnic, la diferite intervale de timp.

Centrele de transfuzie vor avea disponibile prin aplicația Web modalități de înregistrare a intrărilor și ieșirilor de sânge, dar și de modificare a limitelor curente, precum și o interfață de raportare a situației curente.

Aplicația mobilă va pune la dispoziție trei interfețe de utilizator diferite, pe lângă cea de donatori existând și cele de receiveri și de centre de transfuzie, care vor prezenta situația actuală pentru fiecare astfel de utilizator, precum și statistici generate pe perioade de timp (zilnice, săptămânale, lunare, anuale).

Persoanele care se vor înregistra ca pacienți vor putea fi găsite de către donatorii care dețin o grupă sanguină corespunzătoare pentru efectuarea unei eventuale donații și care au declarat același oraș ca fiind cel activ prin intermediul numelui și email-ului declarat la crearea profilului, fiind oferite detalii despre centrul de transfuzie de care aparțin. De asemenea, va fi oferită posibilitatea de a distribui cazul solicitantului în social media, pentru a atrage atenția asupra sa.

Donatorii vor dispune de o serie de informații în funcție de grupa de sânge pe care o au (cui pot dona, cum, de la cine ar putea primi), vor avea posibilitatea completării unui test înainte de a putea dona în cazul în care nu cunosc în totalitate procesul pentru a fi informați corespunzător de condițiile cu care vine acesta și vor avea un istoric al donațiilor pe baza căruia pot beneficia de avantajele oferite de legea în vigoare.

Aplicația va oferi posibilitatea programării pentru efectuarea analizelor sau a recoltărilor de sânge direct din aplicație (atât prin aplicațiile de mail/telefon ale dispozitivului mobil, cât și printr-un modul prezent în aplicația propriu-zisă).

Profilul donatorului va include, pe lângă istoricul donațiilor, informații despre analizele efectuate în vederea verificării condițiilor de sănătate a acestuia.

În cazul în care donatorii nu si-au efectuat încă analizele, aceștia vor fi informați în aplicație de acest lucru și vor fi îndrumați să le realizeze.

În cazul în care rezultatul analizelor donatorilor nu le permite acestora să doneze, aplicația le va permite totuși să distribuie în social media aplicația și/sau cazuri ale receiverilor.

Pentru ușurarea procesului de confirmare a stării analizelor donatorilor, aplicația mobilă va permite încărcarea unui fișier PDF standard, pe baza căruia vor fi preluate informațiile relevante și actualizate în baza de date.

După realizarea efectivă a unei donații, aceasta va fi confirmată de către centrele de transfuzie din interfața Web, la momentul înregistrării unei intrări de sânge existând și posibilitatea indicării donatorului.

În plus, redirecționarea unei ieșiri de sânge de către centrele de transfuzie către un pacient deja existent va fi posibilă la momentul completării formularului din aplicația Web.

Aplicația va cere utilizatorilor permisiunea de folosire a sistemului de localizare a telefonului pentru a putea identifica mai ușor centrele de transfuzie din apropiere pe Google Maps și pentru a obține direcții către acestea. În caz contrar, pe Google Maps vor fi dispuse centrele de transfuzie, dar nu va exista posibilitatea localizării donatorului.

De asemenea, întrucât aplicația efectuează constant requesturi către server, utilizatorii acesteia vor fi nevoiți să aibă acces la o conexiune de internet.

În plus, pentru a fi posibilă încărcarea fișierelor de tip PDF, utilizatorii vor trebui să-și ofere acordul aplicației pentru ca aceasta să poată avea acces la memoria internă/externă a telefonului.

## 2.2 Diagrame UML

### 2.2.1 Diagrame ale cazurilor de utilizare

În cadrul acestui subcapitol vor fi reprezentate diagramele generale ale cazurilor de utilizare, corespunzătoare fiecărei platforme pe care urmează să fie disponibilă aplicația.

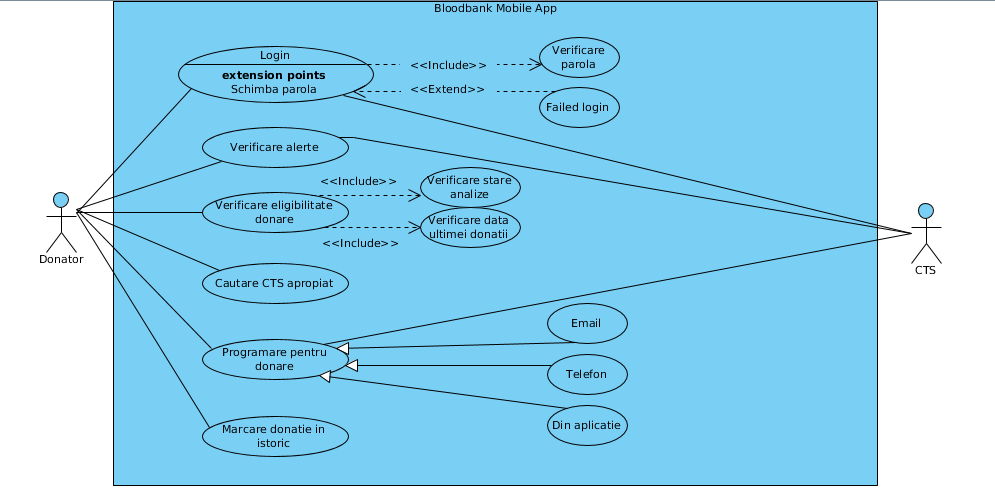


Figura 2.1 - Use Case aplicație mobilă pe tipuri de utilizatori

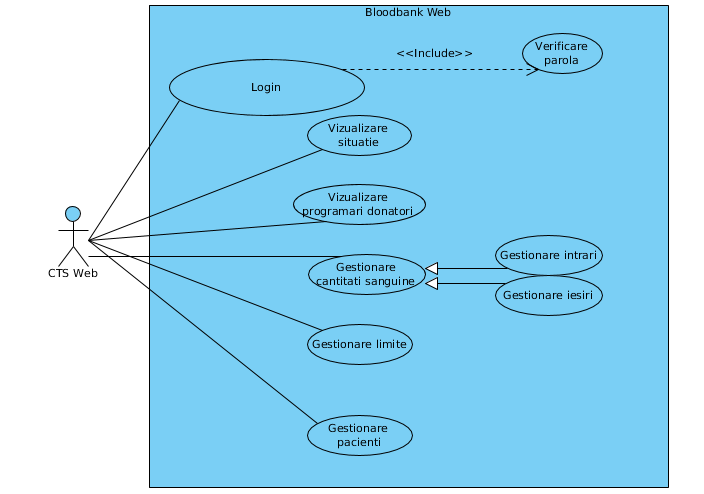


Figura 2.2 Use Case aplicație Web

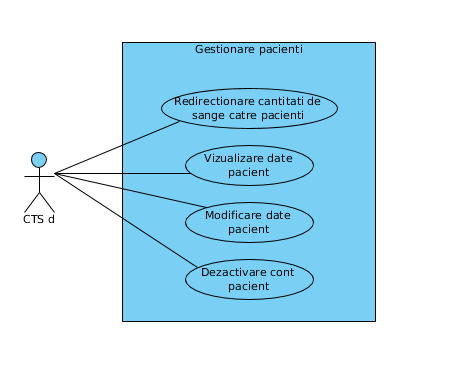


Figura 2.3 Diagrama detaliată a cazului de utilizare a gestonării pacienților de către centrele de transfuzie

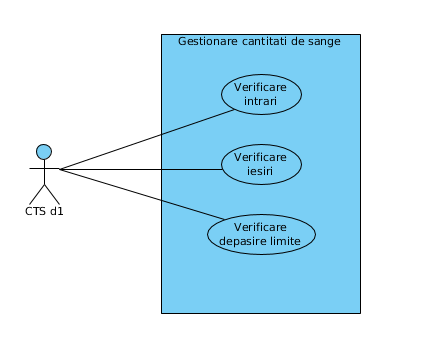


Figura 2.4 Diagrama detaliată a cazului de utilizare de gestionare a cantităților sanguine din cadrul centrelor de transfuzie

|  |  |
| --- | --- |
| Element al cazului de utilizare | Descriere |
| Cod | CU10 |
| Stare | Schiță |
| Scop | Efectuarea unei programări în vederea efectuării donației |
| Nume | Programare pentru donație |
| Actor principal | Donator |
| Descriere | Donatorului îi este oferită posibilitatea de a efectua o programare prin email, telefon sau direct din aplicație |
| Precondiții | Donatorul este logat și este eligibil pentru a dona |
| Postcondiții | Donatorul a efectuat cu succes o programare |
| Declanșator | Donatorul dorește să efectueze o donație |
| Flux de bază | 1. Donatorul introduce data și ora la care dorește să se programeze 2. Donatorul verifică disponibilitatea combinației acestora [Curs alternativ A, B] 3. Donatorul efectuează programarea în cazul în care este posibil |
| Fluxuri alternative | A: Donatorul reia procesul de la pasul 1  B: Se încheie scenariul |
| Relații | Părinte al cazurilor de utilizare de programare telefonică, prin email sau direct din aplicație |
| Frecvența utilizării | Frecvent |
| Reguli ale afacerii | - |

Figura 2.5 Descrierea textuală a CU de efectuare programare

|  |  |
| --- | --- |
| Element al cazului de utilizare | Descriere |
| Cod | CU19 |
| Stare | Schiță |
| Scop | Introducerea “tranzacțiilor” de cantitate sanguină |
| Nume | Gestionare cantități sanguine |
| Actor principal | CTS |
| Descriere | Centrelor de transfuzie le este oferită posibilitatea de a înregistra cantitățile de sânge primite și oferite. |
| Precondiții | Centrele de transfuzie sunt logate  Cantitățile disponibile de sânge nu devin negative |
| Postcondiții | Parcursul cantităților de sânge a fost marcat. |
| Declanșator | Se efectuează o donație/Un pacient primește sânge |
| Flux de bază | 1. Se alege cantitatea de sânge 2. Se alege grupa sanguină de care aparține 3. Se specifică donatorul și/sau pacientul [Curs alternativ: A] 4. Se înregistrează tranzacția |
| Fluxuri alternative | 1. Nu se specifică niciun donator și/sau niciun pacient |
| Relații | Părinte al cazurilor de utilizare gestionare intrări și gestionare ieșiri |
| Frecvența utilizării | Frecvent |
| Reguli ale afacerii | - |

Figura 2.6 Descrierea textuală a CU de gestionare a cantităților de sânge

### 2.2.2 Diagrame de activitate

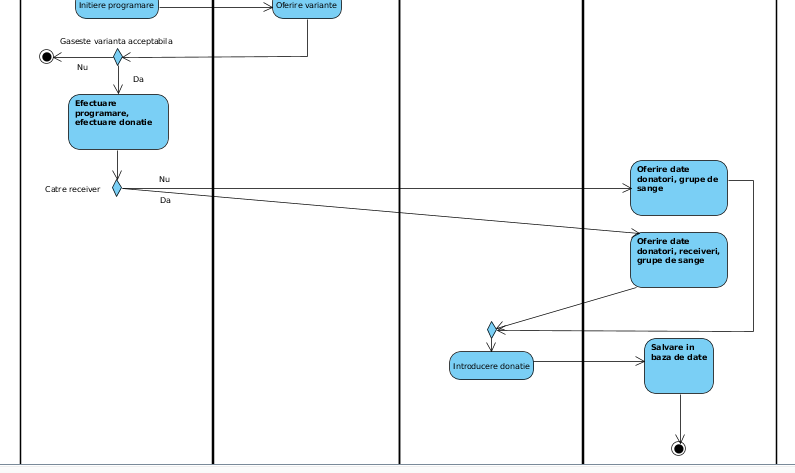
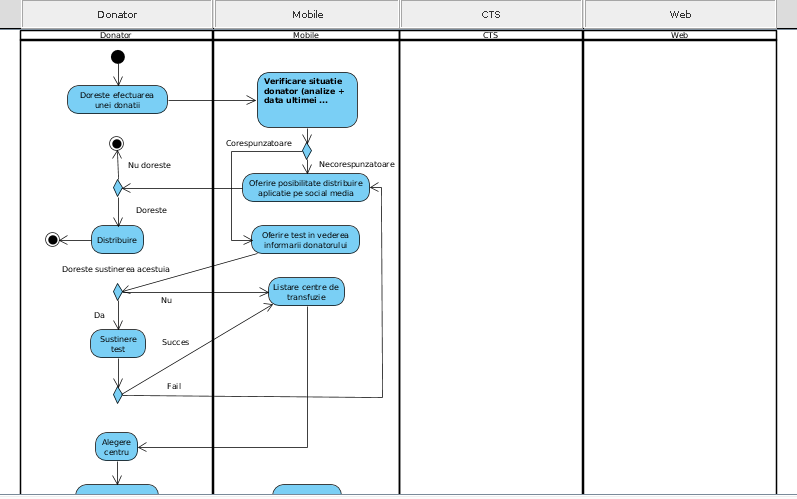


Figura 2.7 Diagrama de activitate a procesului de donare

Această diagramă prezintă modul în care interacționează utilizatorii cu aplicația in realizarea obiectivului propus.

### 2.2.3 Diagrama de clase

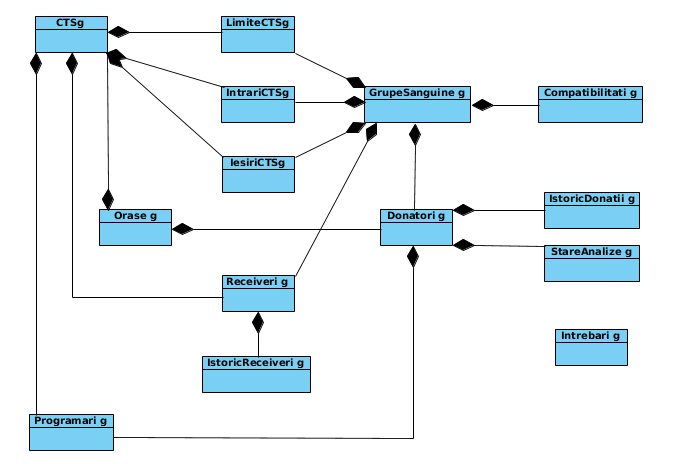


Figura 2.8 Diagrama de clase

Figura 2.7 prezintă atât structura claselor, cât și modul în care acestea interacționează în vederea funcționării corecte a sistemului informatic.

Pe baza relațiilor de agregare compusă ilustrate, se pot identifica principalii “actori” angrenați în procesul general, precum:

* + - Donatorii
    - Receiverii
    - Centrele de transfuzie
    - Grupele de sânge

### 2.2.4 Diagrame de stări

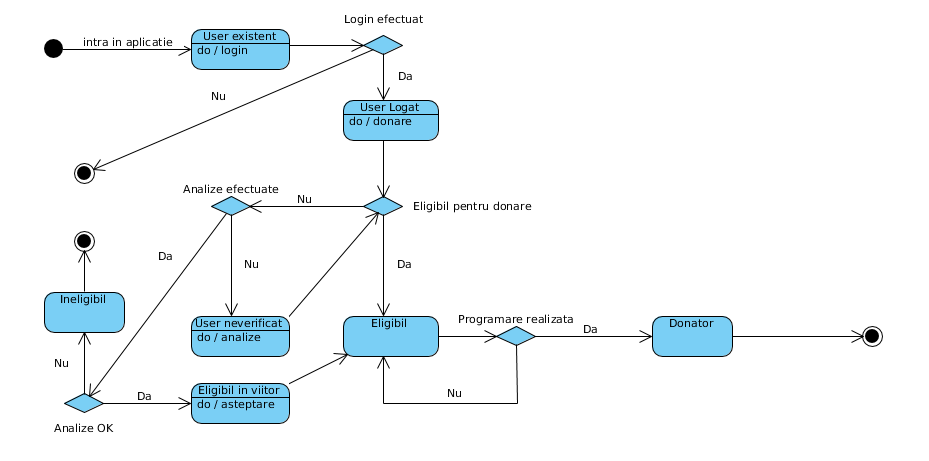


Figura 2.9 Diagrama de stare a donatorilor

Această diagramă ilustrează procesul prin care un user existent trebuie să treacă pentru a putea fi eligibil pentru efectuarea donațiilor.

Inițial, un utilizator de tip donator va avea o stare a analizelor de tip “neefectuate”, această stare urmând a fi actualizată de centrele de transfuzie prin intermediul aplicației Web.

În cazul în care nu există complicații cu analizele utilizatorului, acesta va deveni eligibil pentru efectuarea donațiilor.

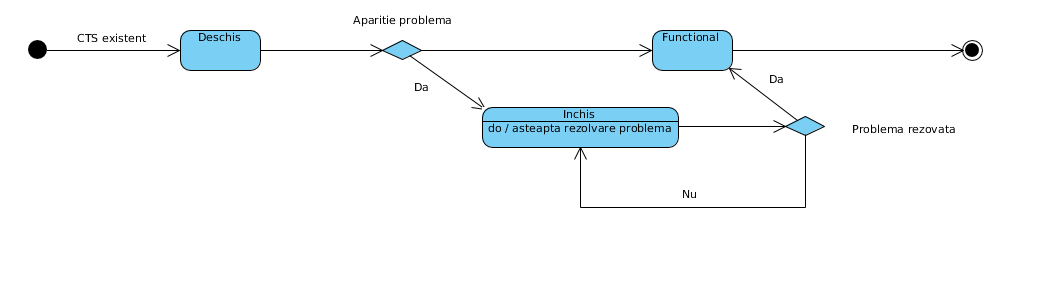


Figura 2.10 Diagrama de stare a centrelor de transfuzie

# CAPITOLUL 3. PROIECTAREA SISTEMULUI INFORMATIC

## **3.1 Rafinarea diagramelor UML**

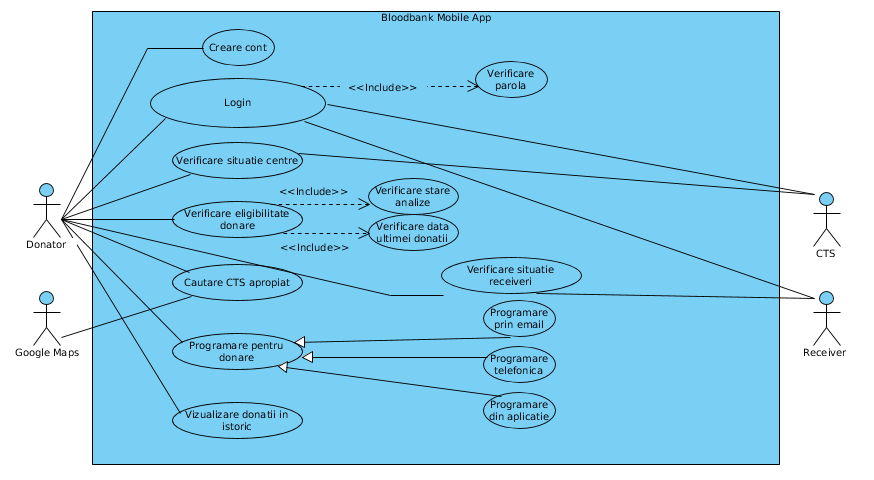


Figura 3.1 Diagrama rafinată a cazurilor de utilizare pentru aplicația mobilă

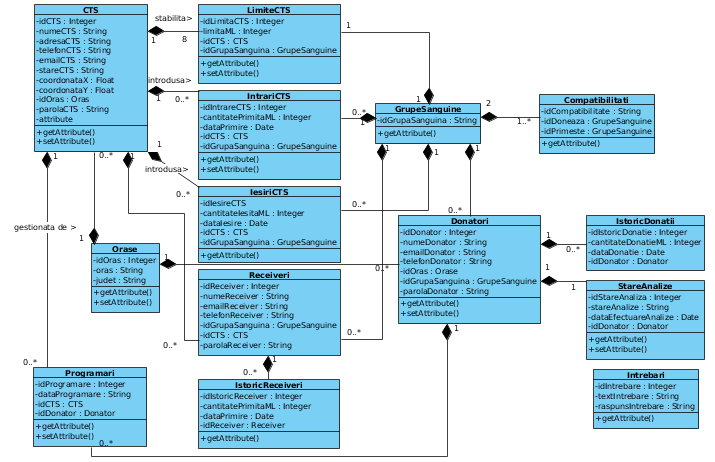


Figura 3.2 Diagrama detaliată a claselor

## 3.2 Proiectarea bazei de date

Primul pas realizat în construirea bazei de date este acela de identificare a claselor pe baza cerințelor funcționale abordate și a situației din viața cotidiană.

Astfel, prin abordarea paradigmei **entitate-asociere** identificăm următoarele entități, grupate pe baza relațiilor dintre acestea, descrise ulterior:

* Orașe
* Grupe sanguine
* Compatibilități ale grupelor sanguine
* Donatori
* Stare analize donator
* Istoric donații
* Centre de transfuzie sanguină (CTS)
* Intrări pe grupe de sânge ale CTS
* Ieșiri pe grupe de sânge ale CTS
* Limite pe grupe de sânge ale CTS
* Programări
* Receiveri
* Istoric receiveri
* Întrebări

Următoarea etapă a acestui proces constituie identificarea asocierilor dintre entitățile identificate anterior, stabilind cardinalitățile existente între acestea:

* Un donator poate avea în istoricul său nicio donație, una sau mai multe, în timp ce un element de tip istoric donații trebuie să aparțină de un singur donator.
* Un donator poate avea cel puțin o înregistrare de tip stare analize, aspect asigurat de un trigger declanșat la crearea unui donator nou, în timp ce starea unei analize trebuie să refere un singur donator.
* Un CTS poate dispune de nicio intrare/ieșire de cantitate de sânge, una sau mai multe, iar un element de tip intrare/ieșire trebuie să refere un CTS.
* Un CTS poate dispune de niciun receiver, unul sau mai mulți, iar un receiver trebuie să aparțină de un CTS.
* Un CTS trebuie să dispună de 8 limite de sânge, aferente fiecărei grupe sanguine, în timp ce o limită de sânge trebuie să aparțină de un singur CTS.
* Un CTS/Donator poate avea nicio programare, una sau mai multe, în timp ce o programare trebuie să depindă de un CTS și de un donator.
* Un oras poate avea niciun CTS/donator, unul sau mai mulți, iar un CTS/donator trebuie să specifice orașul din care fac parte.
* O grupă sanguină poate participa în niciuna, în una sau în mai multe dintre entitățile care se bazează pe aceasta, în timp ce aceste entități trebuie să refere maxim o astfel de grupă.

Din perspectiva paradigmei **bazelor de date**, un alt aspect esențial este identificarea atributelor tabelelor, împreună cu restricțiile de integritate corespunzătoare ale acestora. Toate tabelele conțin o cheie primară individuală de tip Number, cu excepția tabelei GrupeSanguine.

Tabela **GrupeSanguine** are ca și PRIMARY KEY numele efectiv al grupei de sânge, acesta fiind singurul atribut, fiind cunoscute în prealabil toate valorile posibile.

Tabela **Orașe** conține numele orașului, pentru a fi posibilă gruparea CTS, Donatorilor și a Receiverilor.

Tabela **Compatibilități** conține două restricții de tip FOREIGN KEY de tip GrupeSanguine, combinația acestora fiind de tip UNIQUE, fiind cunoscute în prealabil toate compatibilitățile existente.

Tabela **Donatori** conține numele donatorului, precum și email-ul și numărul de telefon al acestuia, pentru a putea fi găsit de centrele de transfuzie în cazul apariției unor complicații. Adresa de email și telefonul au restricții de tip UNIQUE, în timp ce toate câmpurile tabelei sunt de tip NOT NULL.

Tabela **IstoricDonații** conține cantitatea donată în mililitri și data la care a fost efectuată donația. Toate câmpurile tabelei sunt de tip NOT NULL.

Tabela **StareAnalize** conține starea analizelor, de tip CHECK, având trei valori posibile și data la care au fost acestea efectuate. Toate câmpurile tabelei sunt de tip NOT NULL.

Tabela **Receiveri** conține numele receiverului, precum și adresa de email și numărul de telefon al acestuia, pentru a putea fi găsit de donatori în cazul în care aceștia doresc să se intereseze de situația curentă. Email-ul si telefonul au restricții de tip UNIQUE, în timp ce toate câmpurile tabelei sunt de tip NOT NULL.

Tabela **IstoricReceiveri** conține cantitatea primita în mililitri și data la care s-a realizat acest lucru. Toate câmpurile tabelei sunt de tip NOT NULL.

Tabela **CTS** conține numele, adresa de email, telefonul, adresa efectivă a centrului de transfuzie, dar și o stare a acestuia de tip “închis/deschis”, reglementat printr-o restricție CHECK. În plus, pentru dispunerea pe GoogleMaps sunt necesare coordonatele la care acesta se află. Email-ul si telefonul au restricții de tip UNIQUE, în timp ce toate câmpurile tabelei sunt de tip NOT NULL.

Tabela **LimiteCTS** conține cantitatea limită pe grupe de sânge sub care donatorii vor fi notificați de existența unui deficit. Aceasta conține două restricții de tip FOREIGN KEY, CTS și GrupeSanguine, combinația acestora fiind de tip UNIQUE. Toate câmpurile tabelei sunt de tip NOT NULL.

Tabela **IstoricIntrariCTS** conține cantitatea primita în mililitri și data la care s-a realizat acest lucru. Toate câmpurile tabelei sunt de tip NOT NULL.

Tabela **IstoricIeșiriCTS** conține cantitatea ieșită în mililitri și data la care s-a realizat acest lucru. Toate câmpurile tabelei sunt de tip NOT NULL.

Tabela **Programări** conține data pentru care a fost efectuată o programare, această dată fiind de tip UNIQUE. Toate câmpurile tabelei sunt de tip NOT NULL.

Tabela **Întrebări** conține textul întrebării, cât și răspunsul aferent acesteia. Toate câmpurile tabelei sunt de tip NOT NULL.

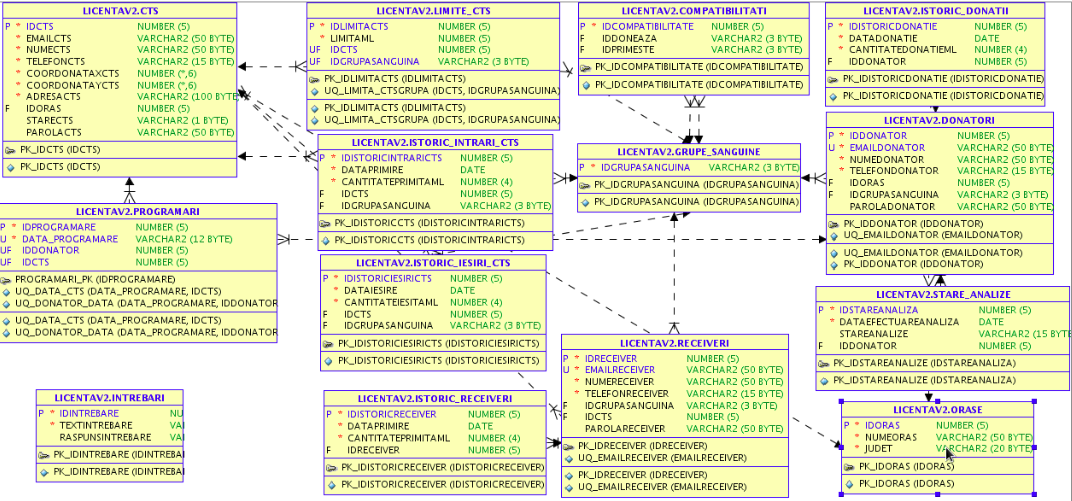


Figura 3.3 Schema conceptuală a bazei de date

## 3.3 Proiectarea ierarhiei de ferestre

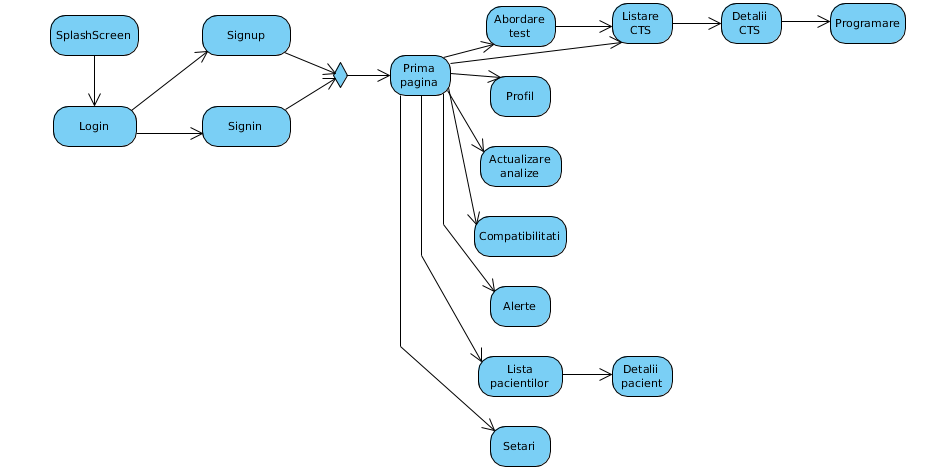


Figura 3.4 Diagrama ierarhiei de ferestre pentru aplicația mobilă

Această diagramă ilustrează modul în care sunt interconectate ferestrele aplicației mobile și fluxul prin care acestea pot fi accesate. Proiectarea acestei scheme a fost necesară în vederea plasării funcționalităților în ferestrele corespunzătoare, astfel încât aplicația să urmeze un curs logic.

## 3.4 Proiectarea interfețelor utilizator

Etapa de proiectare a interfețelor are rolul de a stabili modul în care vor fi dispuse elementele vizuale în cadrul ferestrelor prezentate anterior, în vederea implementării viitoare a funcționalităților propuse. Pentru realizarea acestora, am utilizat programul de realizare de șabloane “Pencil” (<https://pencil.evolus.vn/>).

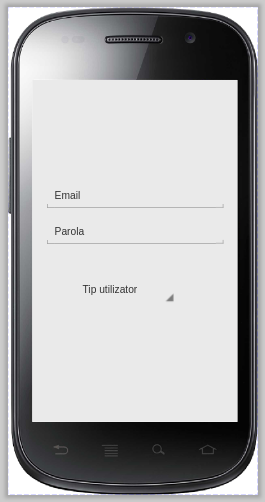


Figura 3.5 Pagina de login Figura 3.6 Fereastra de întrebări



Figura 3.7 Profil pacienți Figura 3.8 Lista clinicilor

## **3.5 Diagrama de componente**

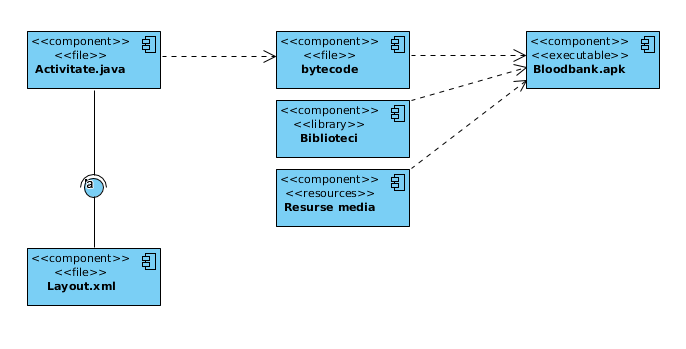


Figura 3.9 Diagrama de componente a aplicației mobile

## **3.6 Diagrama de desfășurare**

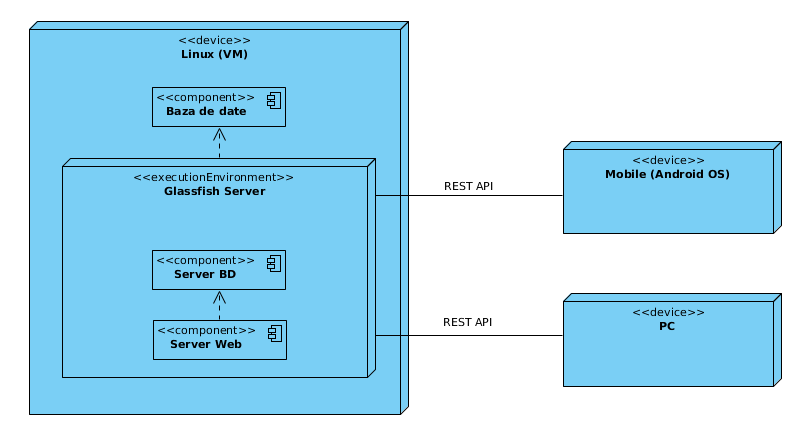


Figura 3.10 Diagrama de desfășurare a sistemului informatic

## 3.7 Tehnologiile informatice utilizate

Pe parcursul dezvoltării acestei soluții informatice va fi utilizată o suită de tehnologii, printre cele mai importante dintre acestea numărându-se:

* Limbajul SQL
* Serviciile REST
* Formatele de reprezentare XML/JSON
* Limbajul Java
* Tehnologiile Servlet și JSP

3.7.1 Limbajul SQL

Primul pas urmat în realizarea aplicației este proiectarea unei baze de date relaționale care să răspundă cât mai eficient cerințelor funcționale ale aplicației, pentru a putea fi utilizată în viitor atât de aplicația de mobil, cât și de cea web.

**SQL** (Structured Query Language) este un limbaj utilizat în descrierea și în manipularea bazelor de date, fiind acceptat de toate sistemele de gestiune a bazelor de date relaţionale [EDU01].

În funcție de scopul urmărit, limbajul SQL pune la dispoziția utilizatorilor 4 mari categorii de comenzi: limbaje de definire a datelor, limbaje de manipulare a datelor, limbajul de control al tranzacțiilor, precum și un limbaj de control al datelor.

În cadrul acestui proiect, limbajul de definire a datelor va fi utilizat pentru a crea, a edita sau a șterge tabelele care descriu datele ce vor fi mai târziu stocate. De asemenea, limbajul de definire a datelor ofera posibilitatea definirii restricțiilor de integritate, restricții ce joacă un rol important în protecția bazelor de date.

Pe de altă parte, limbajul de manipulare a datelor va fi folosit pentru a putea testa corectitudinea structurii bazei de date, prin introducerea, modificarea și ștergerea datelor care vor face obiectul aplicației.

În plus, pentru crearea unei scheme care să permită definirea datelor este nevoie și de implementarea limbajului de control al datelor, fiind necesar un cont cu drepturi de DBA pentru a instanția un user nou.

#### **3.7.2 Serviciile REST**

Următorul pas necesar este reprezentat de expunerea bazei de date, astfel încât aplicațiile client să poată avea acces la informațiile stocate pentru a le putea procesa.

**REST** (Representational State Transfer) este o arhitectură ce permite comunicarea client-server prin intermediul unor formate de reprezentare a datelor. Acest concept este bazat pe protocolul HTTP (HyperText Transfer Protocol), folosind cereri de tip HTTP pentru a putea manipula datele. Serviciul REST pune la dispoziția utilizatorilor metode de tip CRUD (Create Retrieve Update Delete), reprezentate prin verbe (GET, POST, PUT, DELETE etc), pentru a putea accesa/modifica/șterge obiectele unei baze de date, instanțe ce se regăsesc la o adresă specifică la momentul expunerii bazei de date, adresă ce conține un URI (Unique Resource Identifier).

În urma executării unui astfel de request, serviciul REST va returna userului coduri specifice protocolului HTTP specifice rezultatului obținut. Cele mai întâlnite dintre acestea sunt codurile de clasă 2, care indică realizarea cu succes a cererilor, codurile de clasă 4, care semnalează probleme pe parte de client, si codurile de clasă 5, care presupun complicații pe parte de server.

În realizarea acestei arhitecturi a fost utilizat driverul Oracle Thin ca și canal de comunicație cu baza de date, pe partea de client fiind utilizat în cadrul unei aplicații Java, iar pe partea de server acesta deservind rolul de mediator cu baza de date Oracle [NET09]. În urma implementării acestuia, în cadrul mediului de dezvoltare a fost creată o instanță a conexiunii bazei de date aflate la distanță (JDBC - Java Database Connectivity), care mai apoi a fost referită în cadrul procesului de generare a serviciului REST prin intermediul JNDI (Java Naming and Directory Interface). La finalizarea procesului de generare a REST API-ului au fost generate automat metodele de tip CRUD, care mai apoi au fost modificate pentru a putea răspunde cât mai eficient cerințelor funcțiionale ale aplicației.

Formatul de reprezentare utilizat în cadrul efectuării cererilor a fost formatul JSON (JavaScript Object Notation), obiectele din baza de date fiind mapate sub o astfel de structură, care mai apoi va fi parsată de aplicațiile client pentru a le putea instanția cu scopul utilizării lor viitoare.

### 3.7.3 Formatele de reprezentare

Cele mai cunoscute și utilizate formate de reprezentare utilizate în mod curent sunt:

* XML - eXtensible Markup Language
* JSON - JavaScript Object Notation

Ambele formate prezentate anterior au fost create pentru a deservi același scop: stocarea datelor sub o anumită reprezentare în vederea transportului acestora.

În cadrul acestui proiect au fost utilizate ambele formate, însă obiectivul fiecăruia dintre acestea a fost diferit.

Anterior a fost descris serviciul REST și a fost menționat la final formatul utilizat pentru a putea transimite datele de la client la server. Alegerea JSON ca mecanism de transport a fost luată ca urmare a analizării punctelor pro și contra în comparație cu formatul XML. Deoarece formatul JSON este mult mai redus din punct de vedere al dimensiunii este de asemenea și mult mai ușor de parsat de aplicațiile client, spre deosebire de XML, format ce conține mult mai multă informație, a cărui parsare poate fi mult mai costisitoare din punct de vedere al resurselor utilizate [NET10][NET11]. Astfel, se poate afirma că în spatele luării deciziei de folosire a JSON ca mijloc de transport a datelor a stat motivația obținerii unei aplicații cât mai eficiente și cât mai rapide, acesta fiind un pas important în optimizarea acesteia.

Cu toate acestea, a fost utilizat și formatul XML, acesta fiind principalul mod de reprezentare folosit de către aplicațiile Android pentru a putea defini layout-ul activităților, fragmentelor etc. Deși adăugarea elementelor de design poate fi realizat și programatic prin intermediul limbajului Java, fiecare layout anterior menționat necesită referirea unui fișier XML la crearea acestuia. De altfel, design-ul programatic nu este neapărat recomandat, fiind preferată separarea realizării aparenței vizuale a aplicației de logica din spatele funcționării acesteia.

### 3.7.4 Limbajul Java

Limbajul Java este un limbaj de programare orientat obiect, fiind cel mai utilizat astfel de limbaj în mediul IT, conform topului realizat de cei de la TIOBE [NET12]. Acesta a fost dezvoltat în cadrul companiei Sun Microsystems începând cu anii 1990, mai precis de echipa “The Green Team”, condusă de James Gosling, inițial soluția dezvoltată numindu-se Oak. Prima versiune Java disponibilă publicului a fost lansată în anul 1995, aceasta fiind JDK Alpha and Beta. Unul dintre cele mai importante aspecte pe care Java le-a introdus a fost posibilitatea de a executa același program pe mai multe sisteme de calcul, indiferent de arhitectura hardware sau software a acestora (conceptul “*Write Once, Run Anywhere”*), singura condiție fiind prezența unei mașini virtuale cu rol în interpretearea codului Java (JVM).

Tehnologia Java se împarte în patru mari categorii, în funcție de tipul de platformă vizat. În cadrul acestui proiect se regăsesc două dintre acestea, mai exact:

* + - Java Enterprise Edition (Java EE)
    - Java Standard Edition (Java SE)

Java SE este utilizat în dezvoltarea aplicațiilor client, în timp ce Java EE, ca și extensie a Java SE, a fost folosit pentru a crea partea de server care va deservi în viitor atât aplicația mobilă, cât și aplicația web.

### 3.7.5 Tehnologiile Servlet și JSP

Model View Controller (MVC) este un design arhitectural utilizat în dezvoltarea aplicațiilor web, concept ce are ca scop separarea celor mai importante trei componente logice, fiind utilizat pentru a organiza cât mai eficient codul [NET13].

Aplicația dezvoltată, pe parte de Model, cuprinde toate datele transmise în aplicație, fiind utilizate clase mapate pe structura tablelelor din baza de date, care au ca rol preluarea datelor stocate în aceasta. Aceste date vor fi trimise mai apoi componentei de tip Controller, care joacă rolul de intermediar între Model și View. Scopul componentei Controller este acela de a prelucra datele și de a le trimite către View. Ultimul element al acestei arhitecturi, componenta View, are ca obiectiv implementarea logicii interfeței de utilizator a aplicației client (UI) [NET14].

Tehnologiile care permit utilizarea limbajului Java pentru a putea modela un astfel de design sunt: **Servlet și JSP** (Java Server Page), componente oferite prin intermediul Java EE.

Componentele de tip Servlet sunt aplicații Java aflate pe partea de server, deservind rolul de interfață între baza de date și interfața grafică a aplicației, acestea procesând requesturile lansate de către componenta vizuală, preluând datele rezultate în urma executării requestului, pentru ca mai apoi să le trimită înapoi View-ului pentru randare [NET15]. De cele mai multe ori, servlet-urile utilizează protocolul HTTP pentru a procesa cererile [NET16].

Elementele JSP sunt documente text care preiau funcția de afișare a componentelor vizuale, acestea fiind folosite pentru a construi pagini web dinamice. Acestea conțin cod HTML pentru partea statică, în timp ce pentru partea dinamică sunt utilizate concepte specifice JSP, precum JSTL (JavaServer Pages Standard Tag Library) [NET17]. JSTL sunt taguri ce pot fi incluse în JSP, oferind capacități precum definirea structurilor condiționale sau iterative, dar și de preluare a datelor transmise de servlet-uri [NET18].

## 3.8 Produse informatice utilizate

În cursul dezvoltării sistemului informatic, printre produsele informatice folosite s-au numărat următoarele:

* Oracle VM VirtualBox
* SQL Developer (IDE)
* NetBeans (IDE)
* Android Studio (IDE)

IDE (Integrated Development Environment) sunt produse software ce includ funcționalități precum editoare de text, compilatoare și debuggere, oferind o interfață grafică de utilizator (GUI) pentru a facilita întreg procesul de dezvoltare a aplicațiilor [NET19].

### 3.8.1 Oracle VM VirtualBox

Oracle VM VirtualBox este o aplicație desktop ce oferă posibilitatea rulării mașinilor virtuale pe un host, printre care și o multitudine de sisteme de operare de tip guest [NET20].

În cadrul acestui proiect a fost utilizată o imagine a unui sistem de operare Linux, dezvoltat de Oracle, mai exact Oracle Linux Server 6.7, care a venit preinstalat cu SQL Developer și cu un cont de DBA care a permis crearea continuă de scheme și de useri pentru testare, dar și cu o versiune de NetBeans.

### 3.8.2 SQL Developer

SQL Developer este un IDE care integrează SGBD Oracle, permițând executarea de comenzi SQL în funcție de scopul urmărit, oferind și posibilitatea utilizării limbajului PL/SQL.

### 3.8.3 NetBeans

NetBeans este un IDE lansat de către cei de la Sun Microsystems în anul 2000, oferind un mediu de dezvoltare ce se adresează programatorilor specializați în mai multe limbaje, fiind însă cel mai cunoscut ca platformă de dezvoltare Java [NET21].

### 3.8.4 Android Studio

Android Studio este IDE-ul recunoscut oficial ca platformă de dezvoltare a aplicațiilor mobile destinate sistemului de operare Android, fiind dezvoltat de către cei de la JetBrains în coordonare cu cei de la Google. Printre cele mai importante funcționalități pe care le oferă, pe langă cele specifice IDE-urilor în general, se numără și posibilitatea instalării și utilizării emulatoarelor de telefoane Android și nu numai. Acest lucru facilitează testarea programelor dezvoltate pe diferite configurații, atât hardware (dimensiuni, senzori), cât și software (versiuni de API), asigurându-se astfel funcționarea corespunzătoare a acestora.

# CAPITOLUL 4. IMPLEMENTAREA ȘI PREZENTAREA SISTEMULUI INFORMATIC

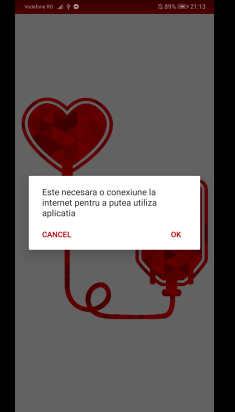
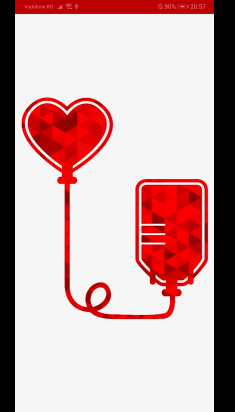


Figura 4.1 Ecranul de start Figura 4.2 Dialog Internet

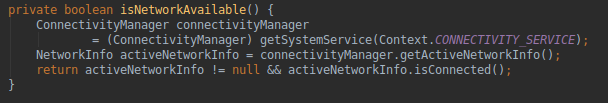


Figura 4.3 Secvența de cod care verifică existența conexiunii la Internet

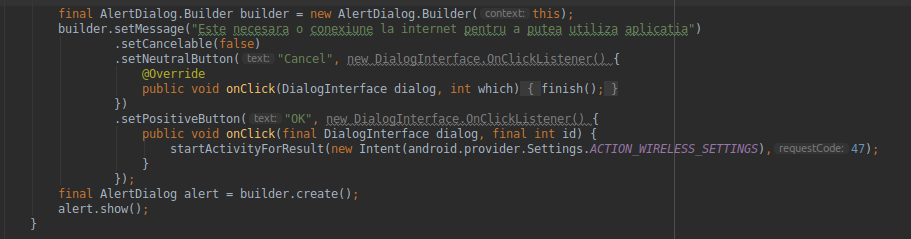


Figura 4.4 Afișarea dialogului de la Figura 4.2

La deschiderea aplicației mobile este dispus ecranul de la Figura 4.1, în care sunt încărcate în memorie anumite elemente necesare funcționării adecvate a acesteia. Pentru acest lucru, este necesară existența unei conexiuni active la Internet, fapt verificat prin codul de la Figura 4.3, iar în cazul în care această condiție nu este îndeplinită, utilizatorul este avertizat prin mesajul din Figura 4.2. Astfel, ecranul de pornire va rămâne dispus până în momentul în care userul va activa conexiunea la Internet, fiindu-i oferită prin opțiunea “OK” a dialogului setarea de Internet din Figura 4.5.

Un alt rol al ecranului de pornire este acela de a verifica dacă un user a fost logat anterior în aplicație, caz în care va fi redirecționat către pagina principală a aplicației, în caz contrar fiindu-i oferită posibilitatea de a se conecta cu un cont sau de a crea unul nou (Figura 4.6). Acest lucru se determină prin referirea fișierelor de preferință ale aplicației, fișiere ce conțin elemente de tip cheie-valoare. Astfel, după crearea unui nou cont sau dupa logarea unui utilizator, în aceste fișiere vor fi salvate informații pe baza cărora va putea fi reconstruit profilul acestuia, informații a căror existență va fi verificată.

În cazul conectării unui user deja existent, acestuia îi vor fi solicitate adresa de email și parola, cât și tipul contului aferent, pe baza căruia se va determina tabela din baza de date asupra căreia se vor efectua operații de regăsire a datelor, precum și pagina principală ce va trebui dispusă (Donator - Figura 4.8, Pacient - Figura 4.45, Centru de transfuzie - Figura 4.28).

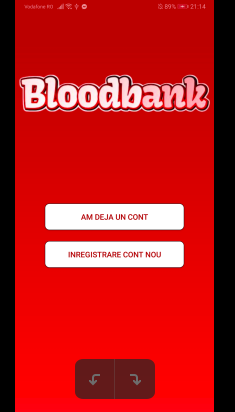
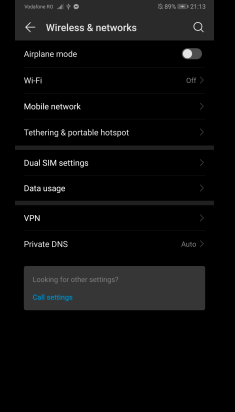


Figura 4.5 Setări Internet Figura 4.6 Alegere signin/signup

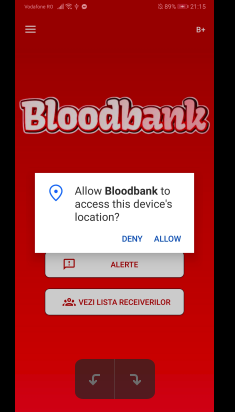
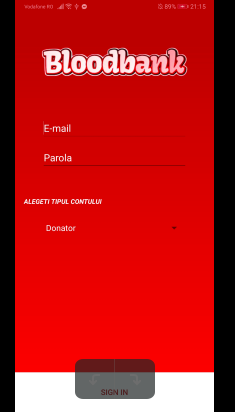


Figura 4.7 Fereastră login Figura 4.8 Cerere permisiuni locație

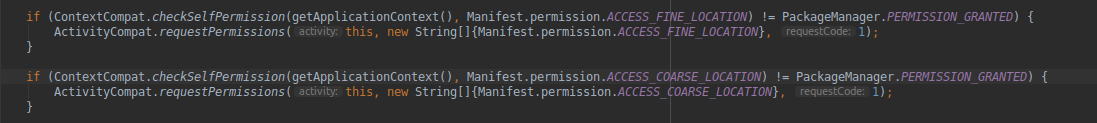


Figura 4.9 Secvența de cod pentru lansarea dialogului de oferire a permisiunilor de locație de la Fig 4.8

În cazul contului de donator, la deschiderea aplicației după prima instalare a acesteia (în cazul în care nu a fost refuzată anterior), va fi cerută permisiunea explicită de a utiliza serviciile de localizare, funcționalitate oferită prin dialogul prezent în Figura 4.8 și implementată prin itermediul secvenței de cod aferente Figurii 4.9. Ulterior, serviciile de localizare vor oferi coordonatele la care se regăsește dispozitivul mobil, acestea fiind utilizate în diefrite module ale aplicației.

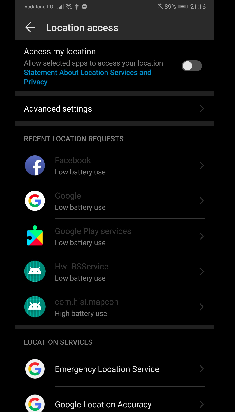


Figura 4.10 Dialog Location Figura 4.11 Setări Location

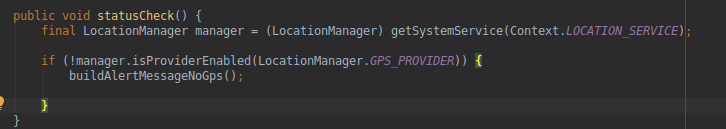


Figura 4.12 Verificarea prezenței serciviilor de localizare

Ca și în cazul conexiunii la Internet, dacă utilizatorul și-a dat acordul pentru utilizarea servicilor de localizare, iar acestea sunt dezactivate, acesta va fi informat și, în cazul în care dorește acest lucru, îndrumat pentru pornirea acestora alegând opțiunea “DA” a dialogului dispus în Figura 4.10. Se vor deschide astfel setările ce țin de serviciile de localizare, de unde pot fi activate. Verificarea funcționării facilităților GPS se realizeaza prin intermediul codului de la Figura 4.12.

După realizarea tuturor acestor pași, pe prima pagină este dispusă grupa aferentă donatorului (Figura 4.8 - colț dreapta sus), cât și elementele grafice ce-i permit acestuia navigarea prin aplicație, fie ele butoane sau elemente ale meniului panoului lateral (Figura 4.12).

În cadrul acestui panou lateral, la evenimentul de click al iconiței din stânga mesajului de întâmpinare a utilizatorului se va putea deschide aplicația de Galerie a telefonului, în care se va putea selecta o nouă imagine care o va suprascrie pe cea curentă cât timp utilizatorul este logat (Figura 4.13 ). Alegerea noii poze se realizează ca urmare a utilizării codului din Figurile 4.14 și 4.15

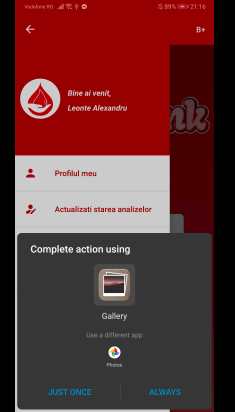


Figura 4.12 Navigation Drawer Figura 4.13 Schimbare poză

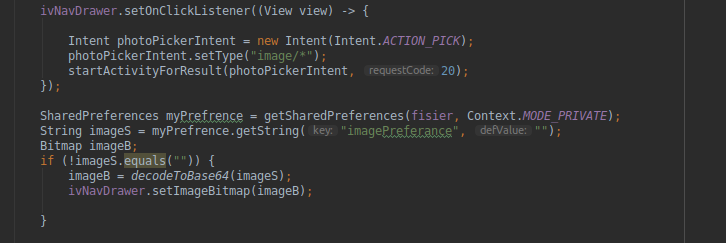


Figura 4.14 Deschiderea galeriei de imagini și afișarea pictogramei selectate

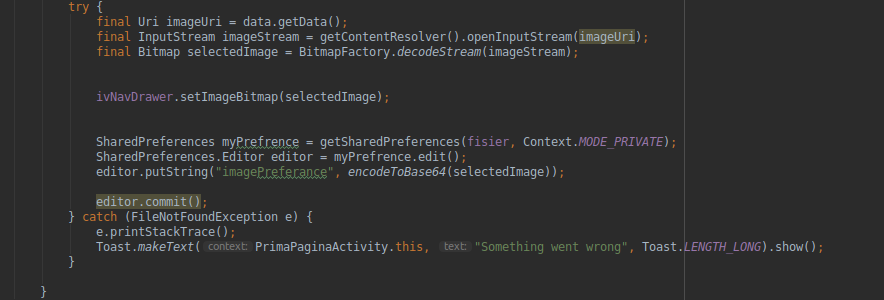


Figura 4.15 Salvarea imaginii preluate din galerie

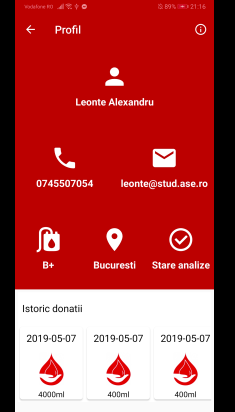


Figura 4.16 Compatibilități Figura 4.17 Profil donator

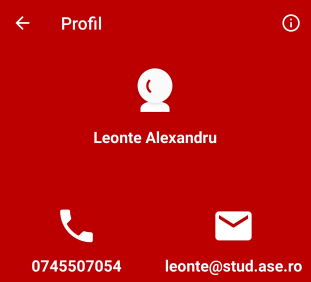


Figura 4.18 Refresh informații profil

În cadrul aplicației există o pagină ce prezintă compatibilitățile grupei curente a utilizatorului (Figura 4.16), cât și o pagină de profil a acestuia, în care sunt prezentate informațiile acestuia, dar și istoricul de donații aferent (Figura 4.17). În cadrul paginii de profil, este posibilă actualizarea datelor prin efectuarea unui swipe vertical, acțiune marcată prin elementul grafic vizibil în Figura 4.18. De asemenea, există posibilitatea deschiderii unui panou de informații, în care este dispusă o legendă a simbolurilor utilizate pentru a descrie starea analizelor a userului.

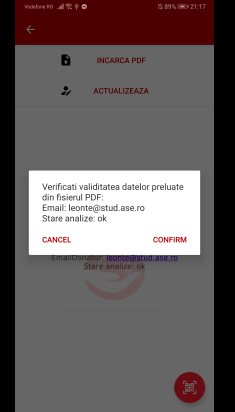
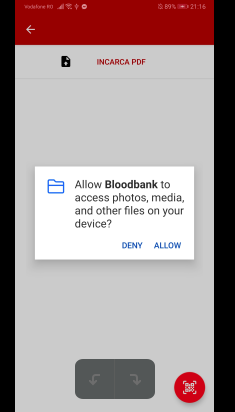


Figura 4.19 Acces memorie Figura 4.20 Confirmare date PDF

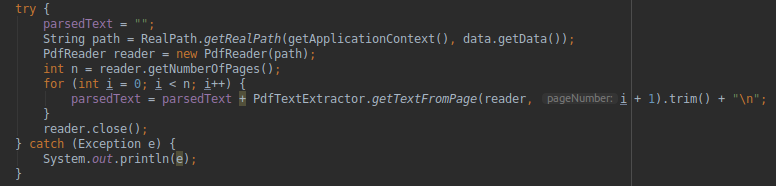


Figura 4.21 Preluare text din PDF

O altă facilitate oferită de aplicație este aceea de a actualiza starea analizelor prin intermediul fișierelor PDF sau a codurilor QR, pentru acestea fiind cerute permisiunile de accesare a memoriei interne și a camerei (Figurile 4.19 și 4.23). În cazul fișierelor PDF, după încărcarea acestora în aplicație, va fi parsat textul conținut în acestea (Figura 4.21) și vor fi căutate elemente specifice pe baza cărora vor fi obținute informațiile necesare. În caz contrar, dar și în cazul în care fișierul PDF nu este destinat utilizatorului curent, acesta va fi informat că fișierul PDF este necorespunzător. Același lucru se realizează și la scanarea codului QR (Figura 4.24), iar dacă procesul decurge în mod normal, utilizatorul va fi informat că actualizarea stării analizelor s-a efectuat cu succes (Figura 4.22).

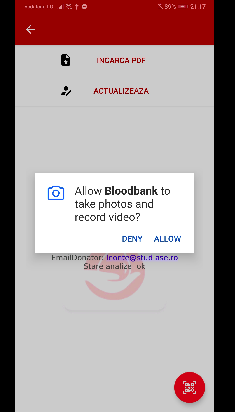
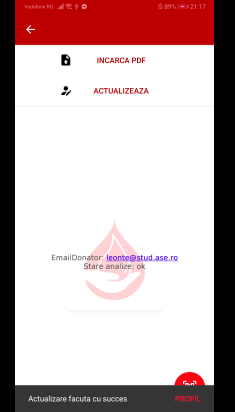


Figura 4.22 Confirmare actualizare Figura 4.23 Acces cameră

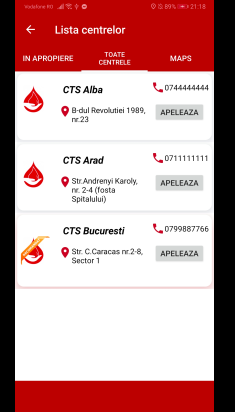
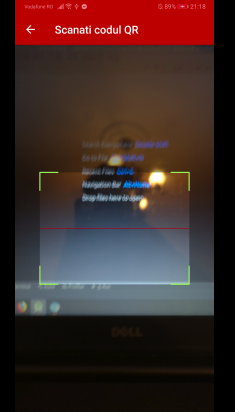


Figura 4.24 Scanare QR Figura 4.25 Lista centrelor de transfuzie cu recomandarea celui mai apropiat pe baza localizării GPS

Una dintre funcționalitățile importante ale aplicației este aceea de a lista centrele de transfuzie sanguină înregistrate în aplicație. Există trei moduri de dispunere a acestora, prima fiind afișarea centrelor existente în orașul utilizatorului, a doua afișând toate centrele de transfuzie, iar cea de-a treia utilizănd serviciul extern Google Maps (Figurile 4.25 și 4.27), fiind adăugate folosind codul de la Figura 4.29.

Astfel, prin folosirea coordonatelor identificate anterior se va putea plasa utilizatorul pe hartă pentru a putea avea o imagine mai clară asupra poziționării centrelor de transfuzie. Totuși, aceasta nu este singura utilitate a coordonatelor userului, acestea fiind folosite și pentru a determina cel mai apropiat centru, aspect ilustrat în Figura 4.25 prin aplicarea stickerului de “recomandat” și realizat prin intermediul codului de la Figura 4.26. Acest lucru poate fi util pentru identificarea celui mai apropiat centru din lista tuturor centrelor sau din lista centrelor din orașul curent.

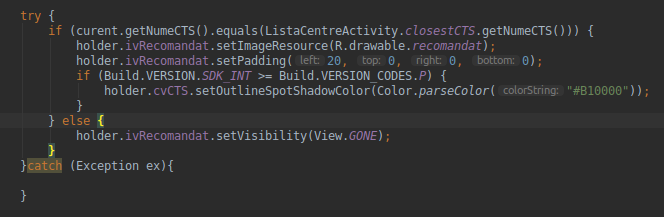


Figura 4.26 Marcarea celui mai apropiat centru de transfuzie

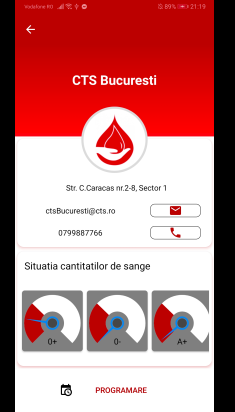
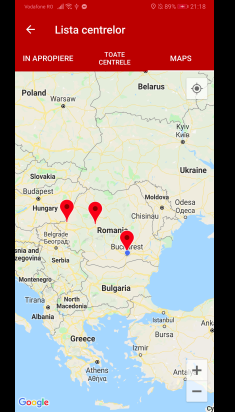


Figura 4.27 Google Maps Figura 4.28 Detalii centre

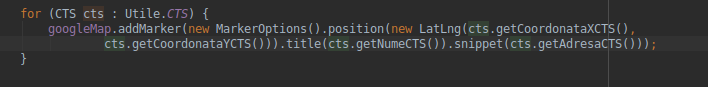


Figura 4.29 Adăugare centru de transfuzie pe hartă

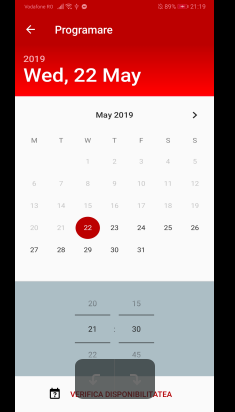
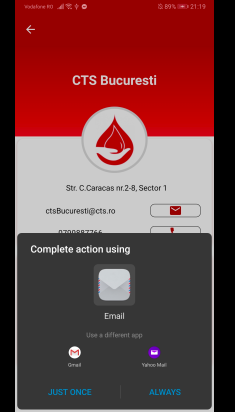


Figura 4.30 Trimitere email Figura 4.31 Pagină programare

Ca urmare a selectării unui centru de transfuzie este dispusă o pagină ce prezintă informații despre acesta, atât detalii de contact precum adresa, numărul de telefon cu opțiunea de apelare sau email (Figura 4.30), cât și informații legate de cantitățile de sânge existente pe fiecare grupă (Figura 4.28). De asemenea, este posibilă efectuarea programărilor în vederea realizării donației, butonul de efectuare propriu-zisă apărând doar dacă data și ora selectate sunt disponibile (Figura 4.32), în caz contrar fiind dispus elementul grafic din FIgura 4.34. După realizarea programării, utilizatorul este informat de acest lucru prin afișarea mesajului de la Figura 4.33.

În cazul în care donatorul are o stare a analizelor corespunzătoare și nu cunoaște condițiile ce trebuiesc îndeplinite pentru a putea dona, acesta poate susține un test pe baza căruia să le afle. În cazul finalizării cu succes a acestuia, îi vor fi dispuse centrele de transfuzie prezentate anterior. În cazul picării testului, precum și în cazul în care starea analizelor nu este cea cerută, acestuia îi vor fi oferite mesaje corespunzătoare, precum și opțiunea distribuirii aplicației în social media.

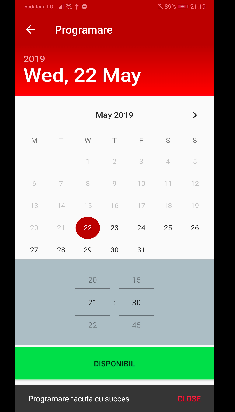
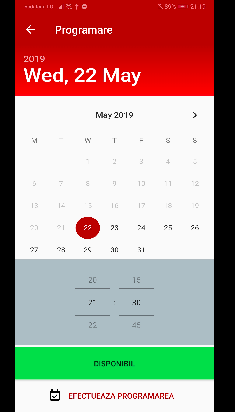


Figura 4.32 Verificare disponibilitate Figura 4.33 Confirmare

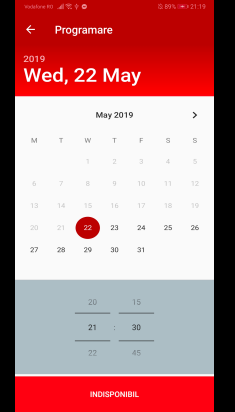


Figura 4.34 Indisponibil Figura 4.35 Dialog susținere test



Figura 4.36 Întrebare Figura 4.37 Răspuns greșit

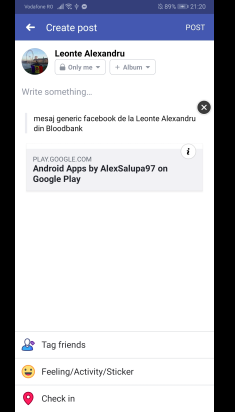
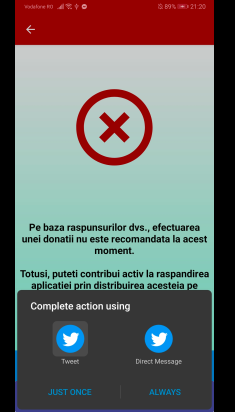


Figura 4.38 Share Twitter Figura 4.39 Share Facebook



Figura 4.40 Deschidere Twitter și Facebook

Figura 4.36 ilustrează interfața specifică fiecărei întrebări, răspunsurile disponibile pentru toate întrebările fiind de tip “Da” sau “Nu”. Aceste întrebări se regăsesc în baza de date, acolo fiind indicate și răspunsurile corecte.

Figura 4.37 dispune unul din multiplele ferestre de eroare, din acestea fiind posibilă distribuirea aplicației pe Twitter (Figura 4.38), respectiv Facebook (Figura 4.39), captarea evenimentelor de click a butoanelor fiind realizată prin intermediul codului de la Figura 4.40. De menționat este faptul că, pentru a avea acces la aceste aplicații, au fost integrate API-urile disponibile pentru fiecare dintre acestea.

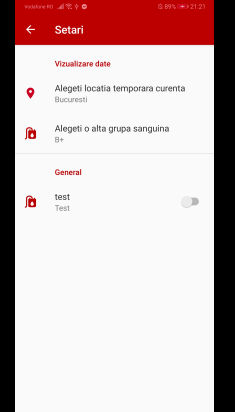
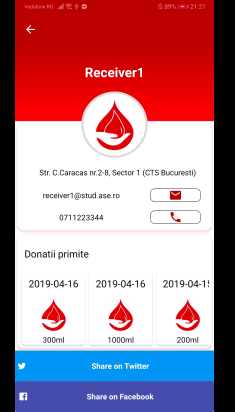


Figura 4.41 Profil pacient Figura 4.42 Setări donator

O altă facilitate importantă este aceea de selectare a pacienților disponibili și compatibili cu grupa de sânge a donatorului în vederea oferirii de informații referitoare la aceștia. Astfel, pe baza Figurii 4.41, putem identifica detaliile de contact ale pacientului, lista donațiilor efectuate către acesta, cât și posibilitatea de a distribui cazul individual pe rețelele de socializare.

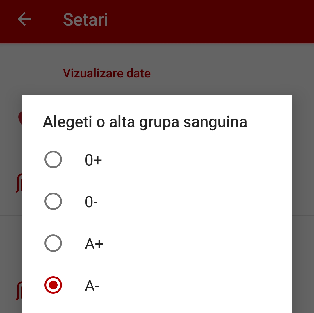


Figura 4.43 Modificare grupă sanguină

kzm_00036

Figura 4.44 Încărcarea grupelor sanguine în lista de la Figura 4.43

Setările aplicației din perspectiva donatorului (Figura 4.42) îi oferă acestuia o serie de opțiuni, printre care modificarea temporară a locației curente, cât și a grupei sanguine. Modificarea acestora nu aduce o modificare și în baza de date, funcționalitatea fiind oferită pentru a putea vedea situațiile și dintr-o altă perspectivă, sau de ce nu, pentru a putea vedea aceste situații din ipostaza unei persoane care vrea să cunoască contextul, dar care nu deține aplicația sau un cont activ la un anumit moment de timp.

Figura 4.43 ilustrează o parte din variantele disponibile pentru schimbarea grupei de sânge, în spatele acestora aflându-se o valoare afișată și o valoare aferentă valorii afișate, în acest caz ele fiind echivalente, precum se poate observa și în Figura 4.44.

În cazul în care contul cu care se realizează autentificarea este unul de tip Receiver, pagina principală va fi cea prezentată în Figura 4.45, pe lângă funcționalitățile întâlnite la pagina din perspectiva donatorului existând posibilitatea de a vedea o statistică a donațiilor primite pe diferite intervale de timp, pagina corespunzătoare fiind dispusă în Figura 4.46. De asemenea, după realizarea obiectivului propus, acela de a dispune de cantitatea de sânge necesară, pacientul are opțiunea de a-și dezactiva contul (Figura 4.47).

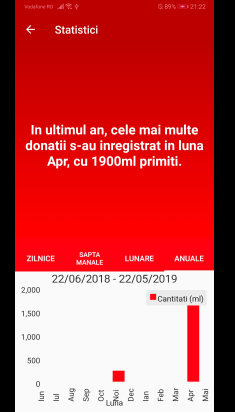
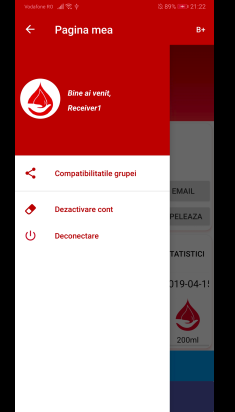


Figura 4.45 Interfață pacient Figura 4.46 Statistici pacient

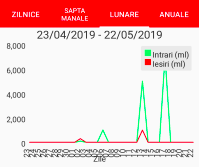
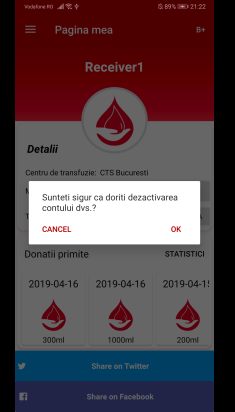


Figura 4.47 Ștergere cont pacient Figura 4.48 Statistici CTS

Ca și în cazul pacienților, centrele de transfuzie dispun de o serie de grafice în care sunt prezentate intrările, respectiv ieșirile de cantitate sanguină pe diferite intervale de timp (Figura 4.48).



Figura 4.49 Notificări donatori

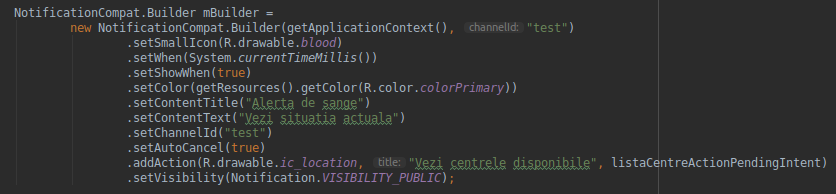


Figura 4.50 Proprietăți notificare

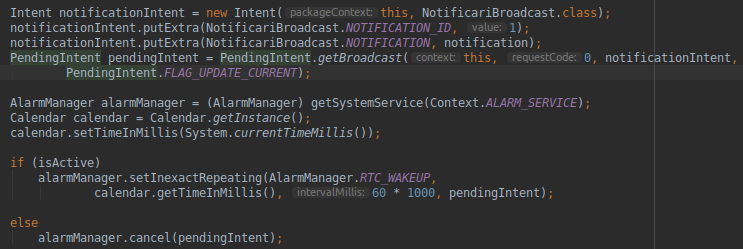


Figura 4.51 Reprogramare notificare

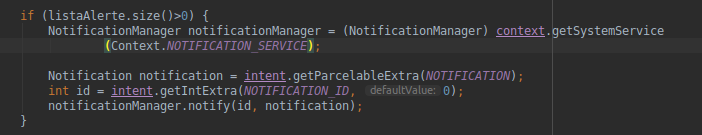


Figura 4.52 Vefificare existență probleme și emitere notificare

O altă funcționalitate ce trebuie menționată este aceea de emitere a notificărilor de fiecare dată când există probleme cu cantitățile de sânge aferente grupelor sanguine compatibile cu cea a donatorului. Acest lucru este realizat în cadrul unui BroadcastReceiver, acest Receiver fiind utilizat pentru a putea fi posibilă lansarea notificărilor chiar și atunci când aplicația nu rulează (Figura 4.52).

Proprietățile notificării, precum mesajul aferent acesteia, pictograma utilizată, culoarea beculețului LED al telefonului, în cazul în care este prezent și altele sunt setate în cadrul codului de la Figura 4.50.

Lansarea efectivă a notificării se va face numai în cazul în care există cel puțin o alertă, acest lucru fiind verificat la intervale regulate de timp cu ajutorul unui AlarmManager în cadrul căruia este setată perioada de timp dorită (Figura 4.51).

Rezultatul tuturor acestor pași se regăsește în Figura 4.49, notificarea oferind posibilitatea dispunerii alertelor, caz în care se vor prelua datele necesare funcționării aplicației într-o activitate transparentă, în cazul în care aplicația nu era pornită.

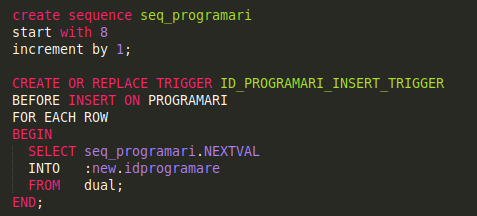


Figura 4.53 Utilizarea unui sequence pentru generarea cheilor primare la declanșarea trigger-ului pe Insert

Întrucât ansamblul informatic efectuează metode de tip POST pentru multiple operații (creare conturi, introducere donație și altele), pentru a asigura o generare corespunzătoare a cheilor primare în vederea inserării în baza de date au fost utilizate o serie de secvențe, dar și de triggeri.

Astfel, la o inserare, în formatul JSON creat spre a fi trimis serverului, valoarea cheii primare nu va fi specificată, ci va fi generată ca urmare a declanșării triggerului la operațiunea de INSERT prin atribuirea acesteia cu următoarea valoare disponibilă a secvenței (Figura 4.53).

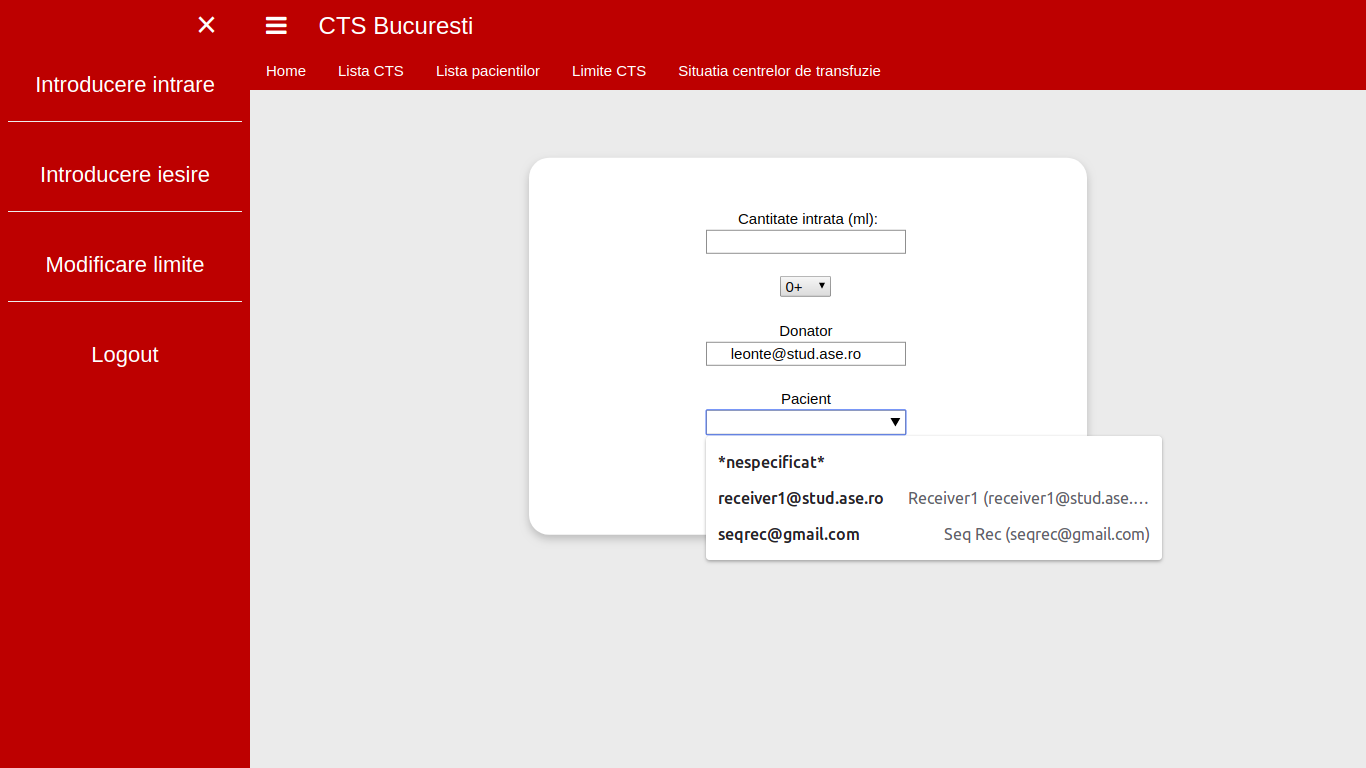


Figura 4.54 Introducerea unei donații de la un donator către un pacient

Platforma Web a aplicației este destinată centrelor de transfuzie în vederea oferirii unui mediu propice inserării datelor. În Figura 4.54 este ilustrată fereastra destinată introducerii unei intrări de sânge, fiind posibilă specificarea unui donator înregistrat în sistem în vederea actualizării istoricului de donații al acestuia, dar și a unui primitor de sânge către care sa fie redirectată ulterior cantitatea de sânge primită. Oricare dintre aceste două câmpuri poate avea valoarea nespecificată. Lista donatorilor include toți donatorii care dețin o stare corespunzătoare a analizelor, în timp ce lista pacienților este formata din aceia care aparțin de centrul de transfuzie logat în aplicație.

În Figurile 4.55-4.60 sunt prezentate mecanismele prin care se verifică corectitudinea valorilor introduse în cazul ieșirilor de sânge, mai exact valoarea cantității de sânge, urmărindu-se ca diferența dintre suma intrărilor și suma ieșirilor, coroborată cu noua cantitate ieșită, să nu fie una negativă.

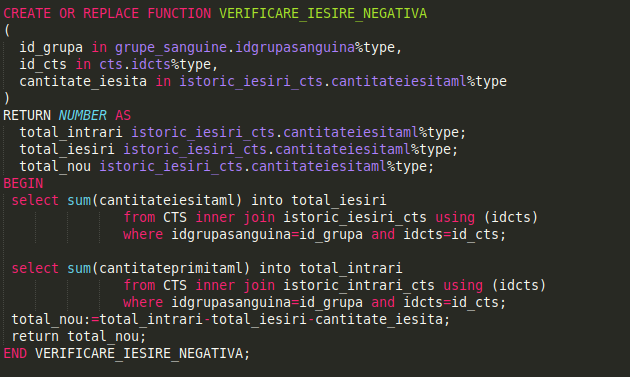


Figura 4.55 Funcția PL/SQL ce returnează noul total al cantității grupei de sânge ce aparține de un anumit centru de transfuzie

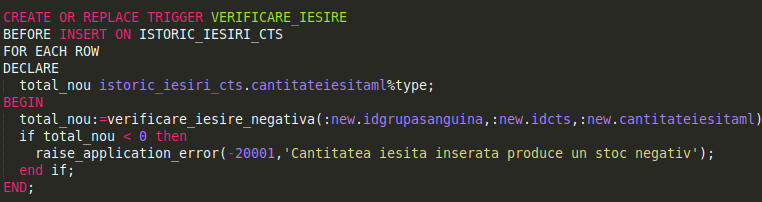


Figura 4.56 Declanșatorul ce împiedică inserarea în baza de date în cazul în care totalul de la FIgura 4.55 este negativ

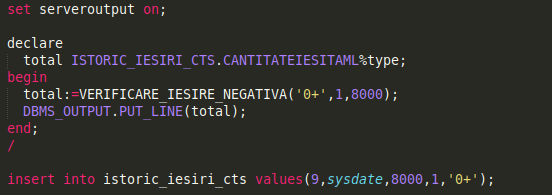


Figura 4.57 Verificarea funcției, respectiv a declanșatorului din Figurile 4.55, 4.56

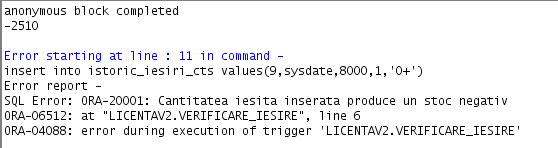


Figura 4.58 Rezultatul rulării codului de la Figura 4.57

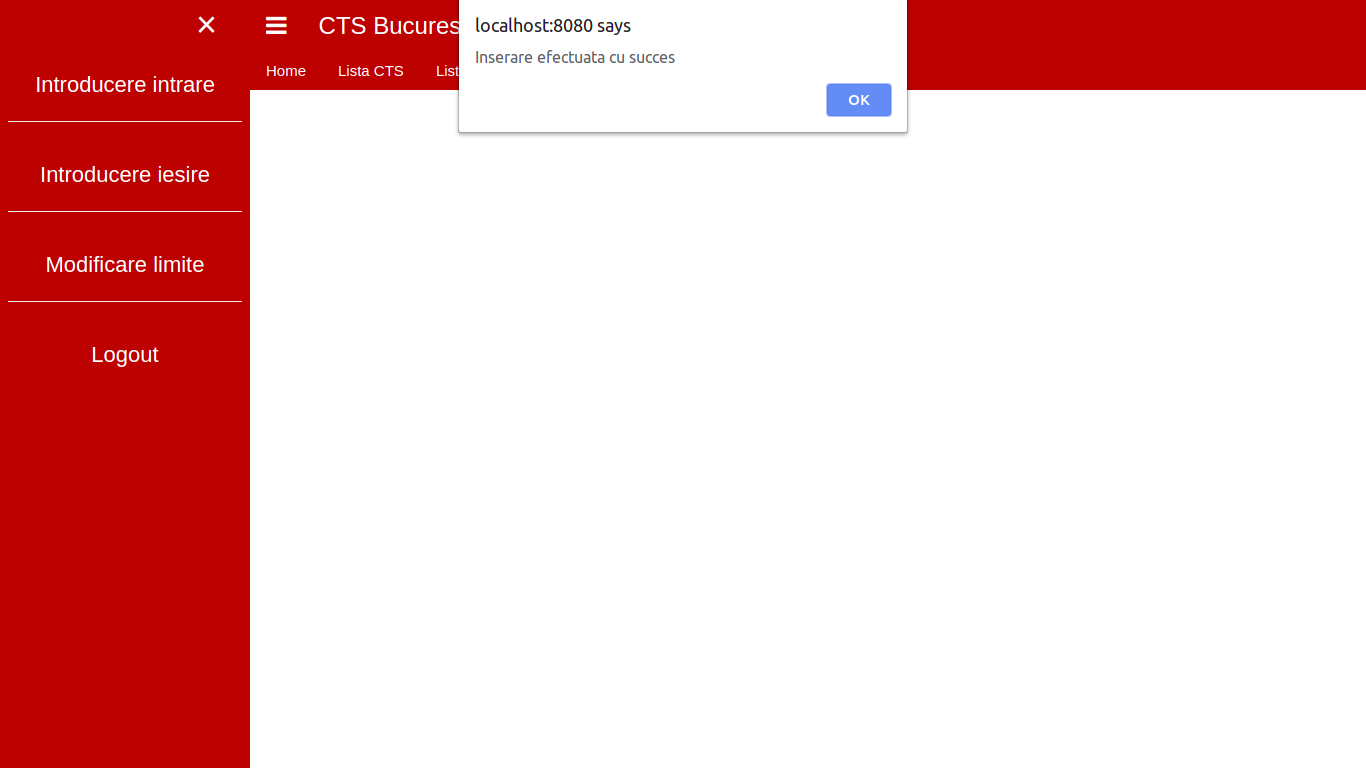


Figura 4.59 Inserarea corespunzătoare a unei ieșiri

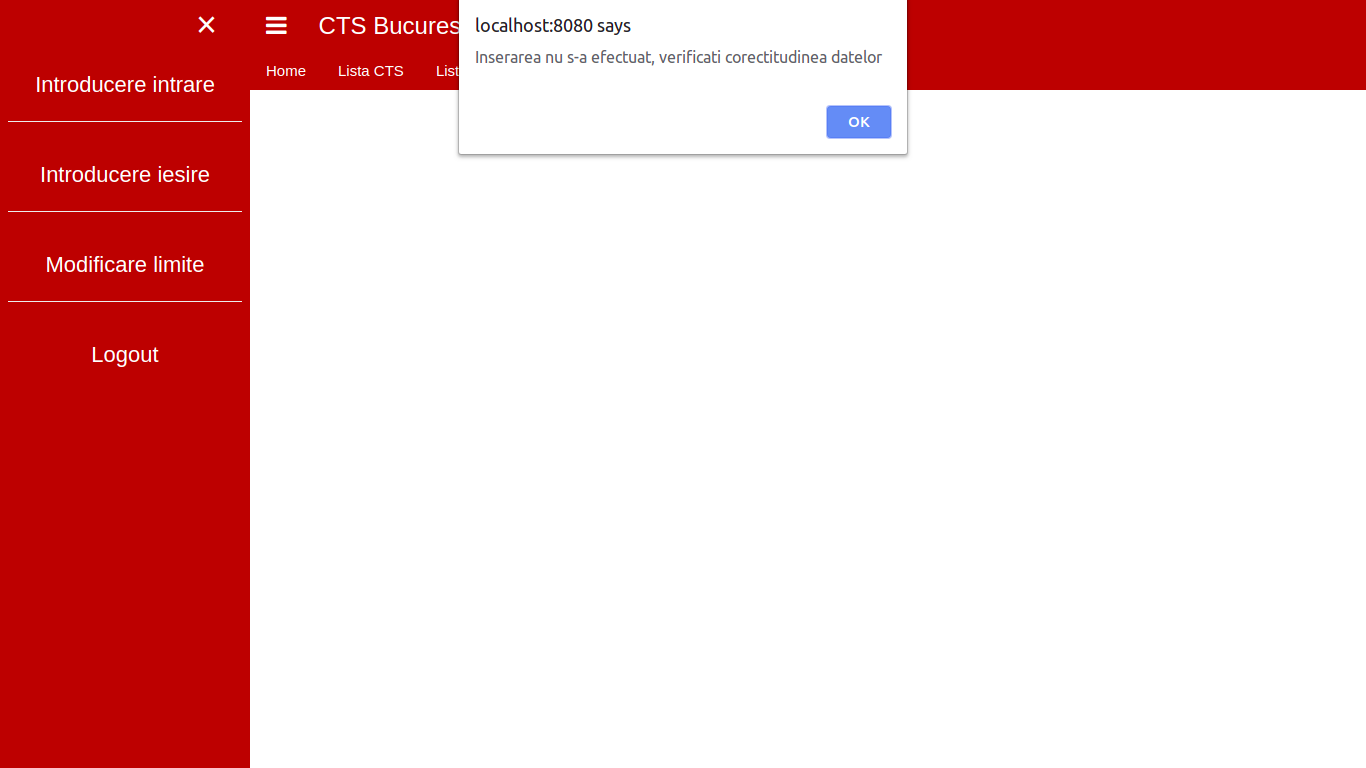


Figura 4.60 Inserarea necorespunzătoare a unei ieșiri

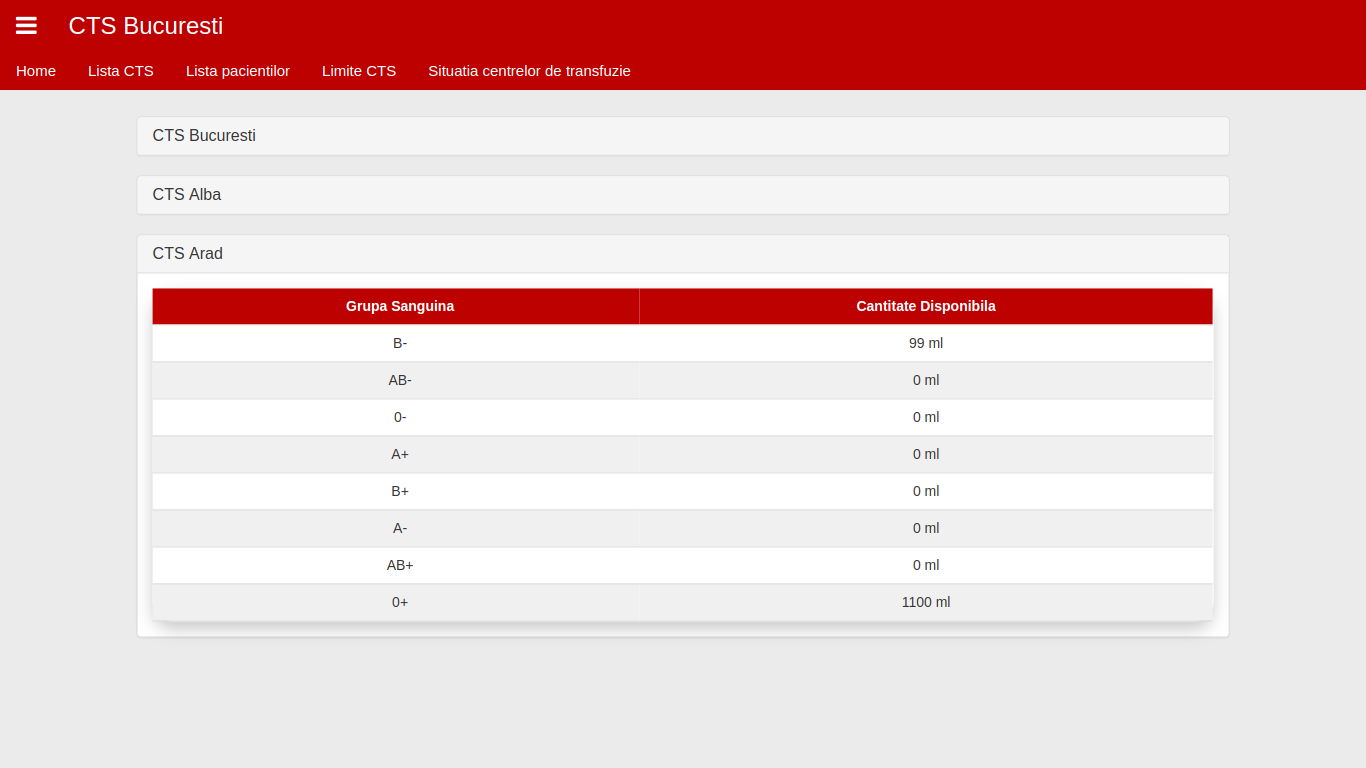


Figura 4.61 Vizualizarea situației tuturor centrelor de transfuzie

# Bibliografie

|  |  |
| --- | --- |
| [POAN05] | POANTĂ LAURA, *Mic dicţionar etimologic de termeni medicali,*  Cluj-Napoca : Casa Cărţii de Ştiinţă, 2005, Bibliogr., ISBN 973-686-731-5 |
| [NOSM13] | Dr Derek Norfolk, Caroline Smith, *Handbook of Transfusion Medicine - 5th edition*, 2013, TSO, ISBN 9780117068469 |
| [DEAN05] | Dean Laura, *Blood Groups and Red Cell Antigens*, Bethesda (MD): National Center for Biotechnology Information (US), 2005.MD 20892-6510 |
| [CRSA14] | Dan Cristescu, Carmen Salavastru, Bogdan Voiculescu, Cezar Th. Niculescu, Radu Carmaciu, *Manual biologie clasa a XI-a*, 2014, Corint, ISBN: 9786068609089 |
| [EDU01] | [Conf.univ.dr. Iuliana Botha (Șimonca)](http://bd.ase.ro/Media/Default/files/team/IB.jpg" \t "/home/alexsalupa97/Documents\\x/_blank), Curs Baze de Date |
| [PDF01] | https://www.isbtweb.org/fileadmin/user\_upload/files-2015/red%20cells/general%20intro%20WP/Table%20blood%20group%20systems%20v4.0%20141125.pdf |
| [NET01] | http://www.isbtweb.org/about-isbt/news/article/new-publication-from-the-red-cell-immunogenetics-and-blood-group-terminology-working-party/ |
| [NET02] | https://www.khanacademy.org/science/high-school-biology/hs-classical-genetics/hs-non-mendelian-inheritance/a/multiple-alleles-incomplete-dominance-and-codominance |
| [NET03] | http://www.rhesusnegative.net/themission/bloodtypefrequencies/ |
| [NET04] | https://www.capital.ro/ministrul-sanatatii-nevoia-de-sange-este-foarte-mare-iar-numar.html |
| [NET05] | https://www.pentruviata.com/de\_ce.html |
| [NET06] | https://www.banca-de-sange.ro/ |
| [NET07] | https://www.curierulnational.ro/donorium-aplicatia-care-sustine-donarea-de-sange-numarul-1-pe-appstore-in-mai-putin-de-24-de-ore/ |
| [NET08] | http://www.proviataoradea.ro/donare-sange/despre-donarea-de-sange.html  (http://www.infoworld.ro/ro/cts-manager/) |
| [NET09] | https://docs.oracle.com/cd/B28359\_01/java.111/b31224/jdbcthin.htm |
| [NET10] | https://www.w3schools.com/js/js\_json\_xml.asp |
| [NET11] | https://blog.cloud-elements.com/json-better-xml |
| [NET12] | https://www.tiobe.com/tiobe-index/ |
| [NET13] | https://www.codecademy.com/articles/mvc |
| [NET14] | https://www.tutorialspoint.com/mvc\_framework/mvc\_framework\_introduction.htm |
| [NET15] | https://www.baeldung.com/mvc-servlet-js |
| [NET16] | https://docs.oracle.com/javaee/6/api/javax/servlet/Servlet.html |
| [NET17] | https://docs.oracle.com/javaee/5/tutorial/doc/bnagy.html |
| [NET18] | https://www.oracle.com/technetwork/java/jstl-137486.html |
| [NET19] | https://www.techopedia.com/definition/26860/integrated-development-environment-ide |
| [NET20] | https://www.virtualbox.org/ |
| [NET21] | https://netbeans.org/about/ |

# Anexe

## Anexa 1. Indexul termenilor utilizaţi

|  |  |
| --- | --- |
| ISBT | [International Society of Blood Transfusion](https://www.isbtweb.org/) |
| Rh | Rhesus |
| CTS | Centru de Transfuzie Sanguină |
| PDF | Portable Document Format |
| UML | Unified Modeling Language |
| CU | Caz de Utilizare |
| SQL | Structured Query Language |
| DBA | DataBase Administrator |
| REST | REpresentational State Transfer |
| HTTP | HyperText Transfer Protocol |
| CRUD | Create Retrieve Update Delete |
| URI | Unique Resource Identifier |
| JDBC | Java DataBase Connectivity |
| JNDI | Java Naming and Directory Interface |
| API | Application Programming Interface |
| JSON | JavaScript Object Notation |
| XML | eXtensible Markup Language |
| Java EE | Java Enterprise Edition |
| Java SE | Java Standard Edition |
| JVM | Java VIrtual Machine |
| JDK | Java Development Kit |
| MVC | Model View Controller |
| JSP | Java Server Page |
| UI | User Interface |
| HTML | HyperText Markup Language |
| JSTL | Java server pages Standard Tag Library |
| VM | Virtual Machine |
| IDE | Integrated Development Environment |
| SGBD | Sistem de Gestiune a Bazelor de Date |
| PL/SQL | Procedural Language/Structured Query Language |
| GPS | Global Positioning System |
| QR Code | Quick Response Code |
| LED | Light Emitting Diode |

## Anexa 2. Indexul figurilor

|  |
| --- |
| Capitolul 1 |
| Figura 1.1 | Componentele pe fiecare grupă sanguină | 6 |
| Figura 1.2 | Reprezentarea genotipurilor pe grupă sanguină | 6 |
| Figura 1.3 | Compatibilitățile grupelor | 7 |
| Figura 1.4 | Structura grupelor sanguine în România | 8 |
| Figura 1.5 | Tabel comparativ | 11 |

|  |
| --- |
| Capitolul 2 |
| Figura 2.1 | Use Case aplicație mobilă pe tipuri de utilizatori |  |
| Figura 2.2 | Use case aplicație Web |  |
| Figura 2.3 | Diagrama detaliată a cazului de utilizare a gestionării pacienților de către centrele de transfuzie |  |
| Figura 2.4 | Diagrama detaliată a cazului de utilizare de gestionare a cantităților sanguine din cadrul centrelor de transfuzie |  |
| Figura 2.5 | Descrierea textuală a CU de efectuare programare |  |
| Figura 2.6 | Descrierea textuală a CU de gestionare a cantităților de sânge |  |
| Figura 2.7 | Diagrama de activitate a procesului de donare |  |
| Figura 2.8 | Diagrama de clase |  |
| Figura 2.9 | Diagrama de stare a donatorilor |  |
| Figura 2.10 | Diagrama de stare a centrelor de transfuzie |  |

|  |
| --- |
| Capitolul 3 |
| Figura 3.1 | Diagrama rafinată a cazurilor de utilizare pentru aplicația mobilă |  |
| Figura 3.2 | Diagrama detaliată a claselor |  |
| Figura 3.3 | Schema conceptuală a bazei de date |  |
| Figura 3.4 | Diagrama ierarhiei de ferestre pentru aplicația mobilă |  |
| Figura 3.5 | Pagina de login |  |
| Figura 3.6 | Fereastra de întrebări |  |
| Figura 3.7 | Profil pacienți |  |
| Figura 3.8 | Lista clinicilor |  |
| Figura 3.9 | Diagrama de componente a aplicației mobile |  |
| Figura 3.10 | Diagrama de desfășurare a sistemului informatic |  |

|  |
| --- |
| Capitolul 4 |
| Figura 4.1 | Ecranul de start |  |
| Figura 4.2 | Dialog Internet |  |
| Figura 4.3 | Secvența de cod care verifică existența conexiunii la Internet |  |
| Figura 4.4 | Afișarea dialogului de la Figura 4.2 |  |
| Figura 4.5 | Setări Internet |  |
| Figura 4.6 | Alegere signup/login |  |
| Figura 4.7 | Fereastră login |  |
| Figura 4.8 | Cerere permisiuni locație |  |
| Figura 4.9 | Secvența de cod pentru lansarea dialogului de oferire a permisiunilor de locație de la Figura 4.8 |  |
| Figura 4.10 | Dialog Location |  |
| Figura 4.11 | Setări Location |  |
| Figura 4.12 | Verificarea prezenței serviciilor de localizare |  |
| Figura 4.12 | Navigation Drawer |  |
| Figura 4.13 | Schimbare poză |  |
| Figura 4.14 | Deschiderea galeriei de imagini și afișarea pictogramei selectate |  |
| Figura 4.15 | Salvarae imaginii preluate din galerie |  |
| Figura 4.16 | Compatibilități |  |
| Figura 4.17 | Profil donator |  |
| Figura 4.18 | Refresh informații profil |  |
| Figura 4.19 | Acces memorie |  |
| Figura 4.20 | Confirmare date PDF |  |
| Figura 4.21 | Preluare text din PDF |  |
| Figura 4.22 | Confirmare actualizare |  |
| Figura 4.23 | Acces cameră |  |
| Figura 4.24 | Scanare QR |  |
| Figura 4.25 | Lista centrelor de transfuzie cu recomandarea celui mai apropiat pe baza localizării GPS |  |
| Figura 4.26 | Marcarea celui mai apropiat centru de transfuzie |  |
| Figura 4.27 | Google Maps |  |
| Figura 4.28 | Detalii centre |  |
| Figura 4.29 | Adăugare centru de transfuzie pe hartă |  |
| Figura 4.30 | Trimitere email |  |
| Figura 4.31 | Pagină programare |  |
| Figura 4.32 | Verificare disponibilitate |  |
| Figura 4.33 | Confirmare |  |
| Figura 4.34 | Indisponibil |  |
| Figura 4.35 | Dialog susținere test |  |
| Figura 4.36 | Întrebare |  |
| Figura 4.37 | Răspuns greșit |  |
| Figura 4.38 | Share Twitter |  |
| Figura 4.39 | Share Facebook |  |
| Figura 4.40 | Deschidere Twitter și Facebook |  |
| Figura 4.41 | Profil pacient |  |
| Figura 4.42 | Setări donator |  |
| Figura 4.43 | Modificare grupă sanguină |  |
| Figura 4.44 | Încărcarea grupelor sanguie în lista de la Figura 4.43 |  |
| Figura 4.45 | Interfață pacient |  |
| Figura 4.46 | Statistici pacient |  |
| Figura 4.47 | Ștergere cont pacient |  |
| Figura 4.48 | Statistici CTS |  |
| Figura 4.49 | Notificări donator |  |
| Figura 4.50 | Proprietăți notificare |  |
| Figura 4.51 | Reprogramare notificare |  |
| Figura 4.52 | Verificare existență probleme și emitere notificare |  |
| Figura 4.53 | Utilizarea unui sequence pentru generarea cheilor primare la declanșarea trigger-ului pe Insert |  |
| Figura 4.54 | Introducerea unei donații de la un donator către un pacient |  |
| Figura 4.55 | Funcția PL/SQL ce returnează noul total al cantității grupei de sânge ce aparține de un anumit centru de transfuzie |  |
| Figura 4.56 | Declanșatorul ce împiedică inserarea în baza de date în cazul în care totalul de la Figura 4.55 este negativ |  |
| Figura 4.57 | Verificarea funcției, respectiv a declanșatorului din Figurile 4.55, 4.56 |  |
| Figura 4.58 | Rezultatul rulării codului de la Figura 4.57 |  |
| Figura 4.59 | Inserarea corespunzătoare a unei ieșiri |  |
| Figura 4.60 | Inserarea necorespunzătoare a unei ieșiri |  |
| Figura 4.61 | Vizualizarea situației tuturor centrelor de transfuzie |  |