**APLICAȚIE WEB ȘI MOBILE PENTRU CABINETE MEDICALE**

LUCRARE DE LICENŢĂ

|  |  |  |
| --- | --- | --- |
|  | Absolvent: | **Paula PĂLTINIȘANU** |
|  |  |  |
|  | Coordonator ştiinţific: | **Conf.dr.ing. Delia-Alexandrina MITREA** |

**2020**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | |  | |
| DECAN, |  | | DIRECTOR DEPARTAMENT, | |
| **Prof. dr. ing. Liviu MICLEA** |  | **Prof. dr. ing. Rodica POTOLEA** | |

Absolvent: **Paula-Paraschiva PĂLTINIȘANU**

**APLICAȚIE WEB ȘI MOBILE PENTRU CABINETE MEDICALE**

1. **Enunţul temei:** *Proiectul își propune elaborarea unui sistem folosit de atât de cabinetele medicale*, cât și de pacienții acestora
2. **Conţinutul lucrării***: Pagina de prezentare, Introducere, Obiectivele Proiectul, Studiu Bibliografic, Analiză și Fundamentare Teoretică, Proiectare de Detaliu și Implementare, Testare și Validare, Manual de Instalare și Utilizare, Concluzii, Bibliografie*
3. **Locul documentării**: Universitatea Tehnică din Cluj-Napoca, Departamentul Calculatoare
4. **Consultanţi**:
5. **Data emiterii temei:** 1 noiembrie 2019
6. **Data predării:** 8 iulie 2020

|  |  |
| --- | --- |
| Absolvent: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  |  |
| Coordonator ştiinţific: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

**Declaraţie pe proprie răspundere privind**

**autenticitatea lucrării de licenţă**

Subsemnatula, **Păltinișanu Paula-Paraschiva**, legitimată cu **Carte de Identitate** seria **AX**, nr. **722678,** CNP **2961011015567**, autorul lucrării **Aplicație web și mobile pentru cabinete medicale**, elaborată în vederea susţinerii examenului de finalizare a studiilor de licență la Facultatea de Automatică și Calculatoare, Specializarea **Calculatoare** din cadrul Universităţii Tehnice din Cluj-Napoca, sesiunea **iulie 2020** a anului universitar **2019-2020**, declar pe proprie răspundere, că această lucrare este rezultatul propriei activităţi intelectuale, pe baza cercetărilor mele şi pe baza informaţiilor obţinute din surse care au fost citate, în textul lucrării, şi în bibliografie.

Declar, că această lucrare nu conţine porţiuni plagiate, iar sursele bibliografice au fost folosite cu respectarea legislaţiei române şi a convenţiilor internaţionale privind drepturile de autor.

Declar, de asemenea, că această lucrare nu a mai fost prezentată în faţa unei alte comisii de examen de licenţă.

In cazul constatării ulterioare a unor declaraţii false, voi suporta sancţiunile administrative, respectiv, *anularea examenului de licenţă*.

|  |  |  |
| --- | --- | --- |
| Data  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | Nume, Prenume  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  |  |  |
|  |  | Semnătura |

**Cuprins**

[Capitolul 1. Introducere – Contextul proiectului 1](#_Toc43793722)

[1.1. Contextul proiectului 1](#_Toc43793723)

[1.2. Domeniul temei alese 3](#_Toc43793724)

[Capitolul 2. Obiectivele Proiectului 5](#_Toc43793725)

[2.1. Specificația proiectului 5](#_Toc43793726)

[2.2. Obiective generale 6](#_Toc43793727)

[2.2.1. Cerințe funcționale 6](#_Toc43793728)

[2.2.2. Cerințe nonfuncționale 6](#_Toc43793729)

[2.3. Motivație 7](#_Toc43793730)

[Capitolul 3. Studiu Bibliografic 9](#_Toc43793731)

[3.1. Introducere 9](#_Toc43793732)

[3.2. Situația aplicațiilor în domeniul medical 9](#_Toc43793733)

[3.3. Tipurile de aplicații din domeniul medical 10](#_Toc43793734)

[3.4. Studiul aplicațiilor existente pentru cabinetele medicale 11](#_Toc43793735)

[3.4.1. icMed 11](#_Toc43793736)

[3.4.2. Medsoft 11](#_Toc43793737)

[3.4.3. Evolvo 12](#_Toc43793738)

[3.4.4. MedXLine 12](#_Toc43793739)

[Capitolul 4. Analiză şi Fundamentare Teoretică 14](#_Toc43793740)

[4.1. Soluția propusă 14](#_Toc43793741)

[4.2. Motivarea soluției 14](#_Toc43793742)

[4.3. Funcționalitatea și rolurile aplicației 14](#_Toc43793743)

[4.4. Tehnologiile și uneltele alese pentru dezvoltarea aplicației 17](#_Toc43793744)

[4.4.1. Java Spring 17](#_Toc43793745)

[4.4.2. HTML5, Bootstrap și React 18](#_Toc43793746)

[4.5. Dependețe externe 18](#_Toc43793747)

[4.5.1. Google Authenticator 18](#_Toc43793748)

[Capitolul 5. Proiectare de Detaliu si Implementare 20](#_Toc43793749)

[5.1. Specificațiile sistemului 20](#_Toc43793750)

[5.2. Arhitectura sistemului 20](#_Toc43793751)

[5.3. Cazurile de utilizare 21](#_Toc43793752)

[5.3.1. Cazurile de utilizare ale angajaților cabinetului 21](#_Toc43793753)

[5.3.2. Cazurile de utilizare ale rolului de pacient 34](#_Toc43793754)

[5.3.3. Cazurile de utilizare ale rolurilor administrative 38](#_Toc43793755)

[5.4. Descrierea interacțiunii cu utilizatorul 42](#_Toc43793756)

[5.4.1. Diagrame de activitate 42](#_Toc43793757)

[5.4.2. Diagrame de secvență 42](#_Toc43793758)

[5.4.3. Diagrame de comunicare 42](#_Toc43793759)

[5.5. Arhitectura bazei de date 42](#_Toc43793760)

[5.6. Decizii de implementare 43](#_Toc43793761)

[5.6.1. Diagrame de clase 43](#_Toc43793762)

[5.6.2. Diagrame de pachete 43](#_Toc43793763)

[5.6.3. Diagrame de componente 43](#_Toc43793764)

[5.6.4. Diagrame de deployment 43](#_Toc43793765)

[Capitolul 6. Testare şi Validare 44](#_Toc43793766)

[6.1. Scenariile testate în cadrul aplicației 44](#_Toc43793767)

[Capitolul 7. Manual de Instalare si Utilizare 46](#_Toc43793768)

[7.1. Manual de instalare 46](#_Toc43793769)

[7.1.1. Cerințe hardware și software 46](#_Toc43793770)

[7.1.2. Inițializarea aplicației 46](#_Toc43793771)

[7.2. Manual de utilizare 47](#_Toc43793772)

[Capitolul 8. Concluzii 48](#_Toc43793773)

[8.1. Analiza rezultatelor obținute 48](#_Toc43793774)

[8.2. Analiză comparativă 48](#_Toc43793775)

[8.3. Dezvoltări și îmbunătățiri ulterioare 49](#_Toc43793776)

[Bibliografie 50](#_Toc43793777)

[Tabelul figurilor utilizate 53](#_Toc43793778)

# Introducere – Contextul proiectului

## Contextul proiectului

În zilele noastre, tehnologia reprezintă o parte esențială pentru viețile fiecăruia, aceasta fiind capabilă să ne ofere o sumedenie de moduri prin care să ne putem facilita viața și să îndeplinim anumite sarcini depunând un minim de efort.

Pentru a susține ideea precedentă, putem considera datele obținite din baza de date statistică de pe portalul Comisiei Economice a Organizației Națiunilor Unite pentru Europa [[1]](#footnote-1)(CEE). Pe portalul organizației se pot efectua interogări pentru a extrage anumite date. Mai jos se regăsesc 3 diagrame corespondente anilor 2017, 2018 și 2019, în care este măsurat procentajul populației (considerând mai multe intervale de vârstă aparținând ambelor sexe) care utilizează internetul pe o bază săptămânală.

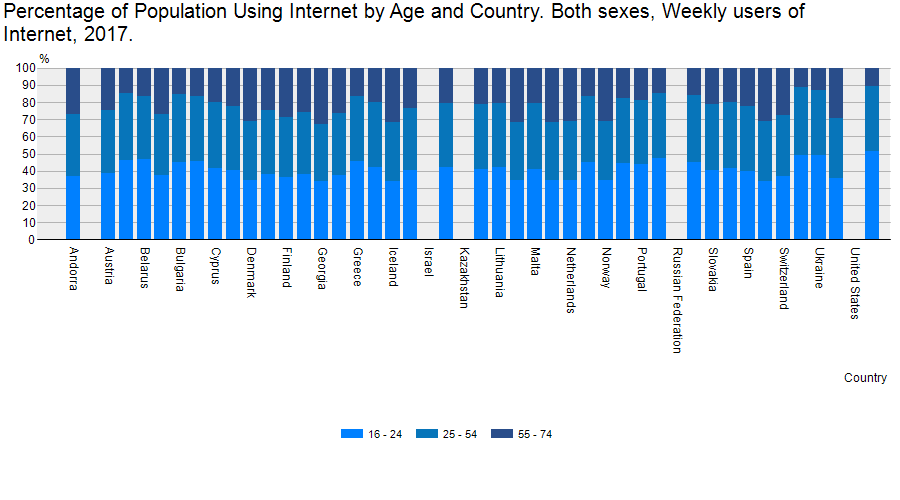


Figura 1.1 Utilizatori săptămânali ai internetului în 2017 (CEE)

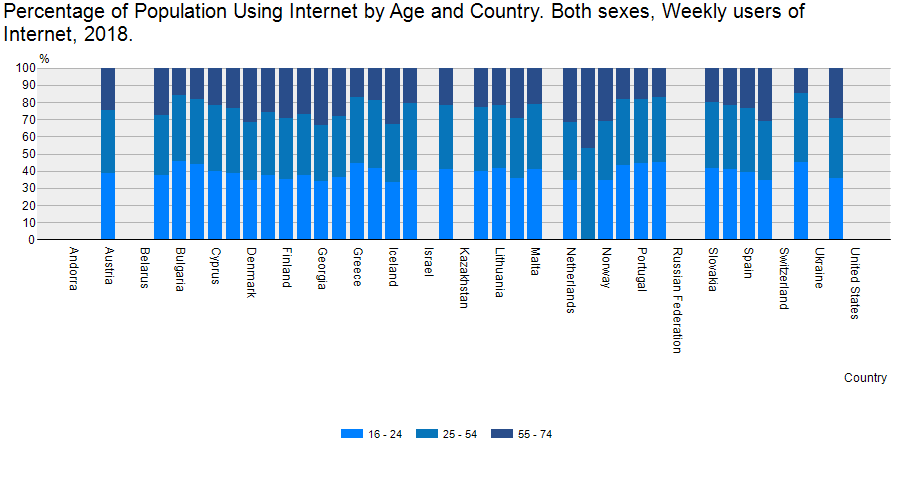


Figura 1.2 Utilizatori săptămânali ai internetului în 2018 (CEE)

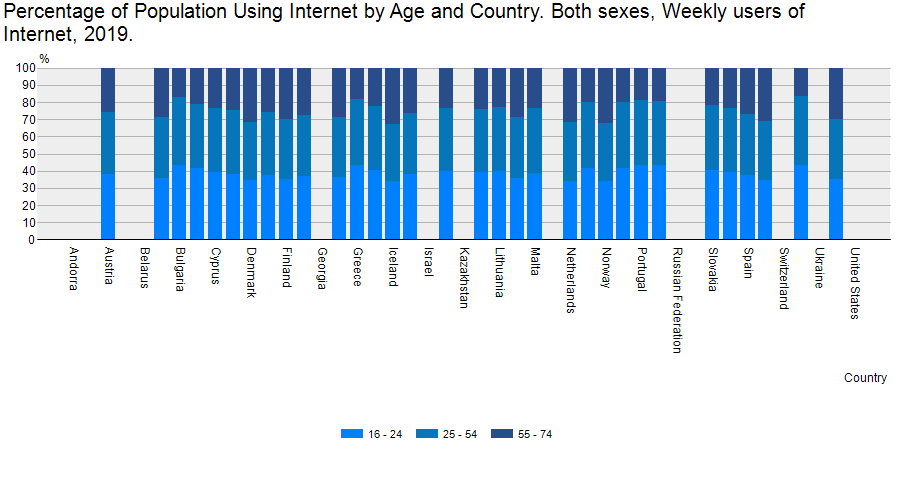


Figura 1.3 Utilizatori săptămânali ai internetului în 2019 (CEE)

Considerând datele din diagrame, putem observa că intervalele de vârstă care utilizează internetul tind să se egalizeze, în sensul că și categoriile superioare de vârstă (55+) încep să utilizeze acest serviciu în aceeași măsură ca și populația mai tânără.

Un studiu [1] realizat de organizația Age UK[[2]](#footnote-2) care are ca obiect utilizarea internetului în rândul persoanelor în vârstă, aduce la iveală că grupul de utilizatori ai internetului a *crescut substanțial în ultimii 5 ani*. Luând fapte în vedere datele expuse anterior, se poate constata că atât numărul, cât și diversitatea aplicațiilor software este într-o continuă creștere, din moment ce numărul utilizatorilor este din ce în ce mai mare, nemaiținând cont de o anumită categorie de vârstă.

Aplicațiile joacă un rol major în viețile noastre, fiind capabile să rezolve probleme din ce în ce mai specifice și detaliate. În cadrul lucrării [2] trei cercetători (T. Soikkeli, J. Karikoski, H. Hämmäinen) ai Departamentului de Comunicare și Networking de la Universitatea Aalto din Finlanda au decis să analizeze *datele a 140 de utilizatori*, pentru *a determina detalii specifice unei sesiuni de folosire a unui smartphone*. Principalele aspecte care au fost luate în calcul au fost acestei cercetări au fost *sesiunile de utilizare a smartphone-ului* și *contexul utilizatorului.* În cadrul acestei cercetări, o *sesiune de utilizare* a fost definită drept *un interval de timp în care o aplicație rulează vizibil (foreground) pe un smartphone*. Legat de contextul utilizatorului au fost definite 5 valori care reprezintă *locurile unde s-au petrecut sesiunile de utilizare ((i) În străinătate, (ii) Acasă, (iii) La birou, (iv) Alte locuri cu însemnătate și (v) Altundeva).* Pentru a colecta datele s-au folosit date primite de la smartphone (*GSM/WCDMA* și *WLAN*), iar rezultatele au fost secționate pe două arii: *diversitatea sesiunilor de utilizare* și *diversitatea contextuală* pentru acestea. Pentru analizarea rezultatelor legate de o sesiune de utilizare, aceștia s-au folosit de *lungimea sesiunii* (interval de timp măsurat în secunde), *numărul de sesiuni într-o durată de timp* (de câte ori a fost lansată o sesiune cu o anumită aplicație), cât și de *timpul de interacțiune cu smartphone-ul*. În urma analizării acestui indice, s-a constatat că *există o diversitate considerabilă în sesiunile de utilizare ale smartphone-ului*, astfel că o sesiune poate varia de la un minim de *50 de secunde* la un maxim de *o oră*. Legat numărul de sesiuni, raza acestora se înscrie în intervalul *3-46 de sesiuni pe zi*, având o durată între *9 minute* și *4 ore și jumătate* pe zi. Analizarea celuilalt indice (sesiunile de utilizare în contextul utilizatorului) a scos la iveală faptul că *aproape jumătate din numărul de sesiuni a avut drept context* mediul de *<<Acasă>>*. O altă informație legată de acest tip de context este faptul că, de obicei, aceste tipuri de sesiuni sunt *cele mai lungi*, cu toate că aceste tipuri de sesiuni *au o frecvență scăzută*. Principalul aspect scos în evidență din această lucrare este faptul că *utilizatorii își folosesc smartphone-urile în mod diferit în contexte diferite*, ceea ce implică faptul că dezvoltatorii de aplicații *ar trebui* *să se conformeze contextului utilizatorului* *pentru a se adapta comportamentelor acestora*.

## Domeniul temei alese

În fiecare zi, cu toții utilizăm sisteme și aplicații software, atât în mod direct – folosind aplicațiile instalate pe calculatorul și/ sau telefonul personal, cât și în mod indirect, ca atunci când accesăm anumite servicii, precum cele ale administrației locale, servicii de curierar, cât și servicii de natură medicală. În ficare dintre aceste domenii ale caror servicii le accesăm zi de zi există o sumedenie de aplicații, dezvoltate pentru diverse platforme (desktop, web, mobile, etc), capabile să eficientizeze procese, făcând posibilă economisirea de timp și de resurse.

La momentul curent, cu toate că accesăm servicii și folosim indirect aplicații în cadrul acestui proces, fiecare persoană are instalate un set de aplicații pe smartphone (sau tabletă) cu care interacționează zi de zi. Aplicațiile pot varia ca domeniu, însa majoritatea persoanelor dețin și aplicații care fac parte din sfera domeniului medical sau de sanatate. Bazat pe un studiu [3] realizat în anul 2015, având ca grup țintă 5 țări (SUA, China, Japonia, Brazilia și Rusia), care a avut drept scop contorizarea și întorcmirea unui sistem de tip “repository” pentru aplicațiile de tipul mHealth, a reușit să obțină o estimare a numărului de aplicații din această sferă: se estima că *numărul aplicațiilor* din de acest era *de peste 100 000* [3], fiind contorizate doar aplicațiile *disponibile în Apple Store și Google Play*. Există, așadar, un număr foarte mare de aplicații în domeniul medical, acestea încadrându-se în sub-domenii variate sau fiind concentrate pe o anumită arie [4].

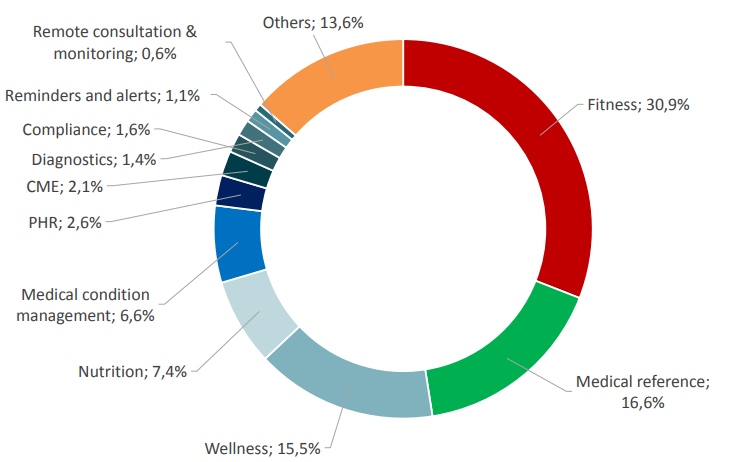


Figura 1.4 Tipurile de aplicații medicale din magazinele online [4]

Diagrama prezentată a fost preluată din studiul [4] prezintă o metrică pentru tipurile de aplicații din categoria *mHealth*; principalele categorii, ordonate descrescător după procente, în care se încadrează aplicațiile sunt: fitness, cu referință medicală (engl. *medical reference* - aplicații care *oferă informații despre medicamente, boli, simptome și sfaturi*), bunăstare (engl. *Wellness -* aplicații care *rezumă soluții de relaxare, instrucțiuni de yoga și sfaturi de înfrumusețare*), altele (engl. Others), nutriție (engl. *Nutrition*), gestiunea unei condiții medicale (engl. *Medical condition management*), fișă de sănătate personală (engl *PHR – Personal Health Record*), educație medicală continuă (engl. *CME* – *Continue Medical Education*), conformitate *(*engl. *Compliance*), diagnosticare (engl. *Diagnostics*), alterte (engl. *Reminders and alerts*), cât și consulturi și monitorizare la distanță (engl. *Remote consultation and monitoring*). Tipurile de aplicații menționate anterior sunt, de cele mai multe ori, disponibile fie pentru angajații centrului medical, fie pentru pacient.

Luând în calcul date din anii 2014 și 2015, toate că numărul [3], dar și varietatea [4] aplicațiilor din domeniul medical arată o valoare ridicată, aceste tipuri de aplicații se încadrează o *piață de nișă*, este estimat că acest lucru *se va schimba în anii următori* [4].

# Obiectivele Proiectului

## Specificația proiectului

Aplicația realizată este, conceptual, secționată în trei module cu funcționalități distincte, fiecare dintre aceste module adresându-se unui anumit tip de rol: administrativ, personal medical sau pacient. Luând în considerare această divizare a rolurilor, considerăm că din rolul de tip ”administrativ” fac parte administratorul de aplicație și cel de cabinet, din rolul de tip ”personal medical” fac parte doctorii și asistenții medicali, iar din rolul de tipul ”pacient” vor face parte strict pacienții cabinetului. Existența acestor roluri facilitează funcționarea unui cabinet ca un tot unitar, fiind separat totodată de interacțiunea cu alte cabinete. Pentru a căpăta o mai bună înțelegere asupra funcționării aplicației în cadrul unui cabinet, se vor exemplifica în continuare etapele prin un cabinet este înregistrat și modul în care procesele sale sunt gestionate în cadrul proiectului implementat.

1. Administratorul de aplicație crează o nouă entitate de tip cabinet, specificându-i informațiile specifice lui, și creează un utilizator de tip ”administrator de cabinet” pentru entitatea creată ulterior
2. Odată creat contul administratorului de cabinet, acesta se poate loga în aplicație și poate crea, la rândul său, utilizatorii de tip ”doctor” și ”asistent medical”
3. Asistenții medicali și doctorii se autentifică în aplicație, având posibilitatea de a înregistra pacienți la cabinet, de a realiza operațiuni folosind datele acestora (programări, consulturi, asignare de prescripții și diagnostice) sau de a genera anumite tipuri de rapoarte
4. Odată ce un cadrul medical înregistrează un pacient, acesta primește access în aplicație, fiind capabil să folosească diverse funcționalități, precum adăugarea și editarea de informații proprii (personale sau medicale), cât și solicitarea de programări către cabinetul medical

Din moment ce toate tipurile de utilizatori există în cadrul unui cabinet, gestiunea proceselor acestuia, dar și a cabinetului propriu-zis este facilitată de existența mai multor tipuri de utilizatori, care pot executa operații la un nivel modular: administratorii de aplicație se pot ocupa doar de datele administratorilor de cabinete și de cabinete în sine, administratorii de cabinet se pot ocupa doar de datele personalului de cabinet și al cabinetului din care fac parte, iar personalul medical are access doar la datele personale proprii, la datele pacienților, cât și la cele anexe operațiunilor medicale (programări, consulturi, rapoarte).

Aplicația se prezintă sub forma unei aplicații web, fapt ce permite un grad ridicat de utilizabilitate și de accesbilitate, singura condiție pentru folosirea acestei aplicații fiind o conexiune stabilă la Internet. Toate tipurile de utilizatori ai aplicației se vor autentifica folosind aceeași pagină și, odată ce aceștia își introduc datele de autentificare corecte, sistemul determină rolul utilizatorului care s-a autentificat și paginile disponibile în cadrul aplicației vor fi cele specifice rolului utilizatorului. Funcționalitățile garantate de aplicație sunt specifice fiecărui rol și pot fi utilizare odată de a avut loc o autentificare cu succes.

## Obiective generale

Principalul obiectiv al realizării acestui proiect este implementarea unei aplicații care să faciliteze și să eficientizeze procesele din cadrul unui cabinet medical. În continuare, vor fi enumerate cerințele funcționale și non-funcționale realizate în procesul de dezvoltare al acestui proiect.

### Cerințe funcționale

În cadrul aceste secțiuni vor fi descrise cerințele funcționale pe care aplicația dezvoltată le îndeplinește.

* Gestiunea unui cabinet medical și al utilizatorilor acestuia (administrație, personal medical, pacienți)
* Gestiunea eficientă a programărilor pentru angajații cabinetului, prin existența unui sistem prin care se poate vizualiza calendarul săptămânal al cabinetului, conținând toate solicitările de programare (pe care doctorii le pot accepta sau refuza), cât și programările deja confirmate (care vor deveni consulturi atunci când pacientul se va prezenta la cabinet)
* Primirea de email-uri: la adăugarea în cabinet (noii utilizatori, atât angajații cabinetului, cât și pacienții, vor primi un email când vor fi adăugați în cadrul cabinetului), modificarea statusului unei programări solicitate (un doctor poate accepta sau refuza solicitarea de programare a unui pacient, moment în care pacientul este notificat), dar și indicațiile către cea mai apropiată farmacie (atunci când unei consultații i se atribuie un diagnostic de către doctor)
* Posibilitatea generării de rapoarte (angajații cabinetului pot genera rapoarte legate de datele medicale ale pacienților, cât și de istoricul programărilor acestora)
* Autentificarea cu o securitate sporită (angajații cabinetului au posibilitatea ca, pe lângă setarea unei parole, să poată folosi un cod de autentificare suplimentar)
* Editarea de informații personale (și, pentru pacienți, a celor de natură medicală – antecedente heredocolaterale, alergii, diagnostice confirmate, etc.)

### Cerințe nonfuncționale

Principalele cerințe nonfuncționale ale aplicației, oferite în speță, de tipul aplicației (web) sunt enumerate și descrise în continuare.

* **Portabilitate**

Aplicația poate fi accesată de pe orice tip de dispozitiv, atâta timp cât se utilizează un browser.

* **Securitate**

Toți utilizatorii care vor dori să acceseze și să utilizeze aplicația vor trebui să se conecteze utilizând propriile acreditări. Pe baza acreditărilor acestora, aceștia vor accesa modulele relevante pentru tipul lor.

* **Mentenabilitate**

Funcționalitățile aplicației sunt implementate în așa fel încât cuplarea este redusă, astfel că, în cazul în care va fi nevoie să se livreze funcționalități ulterioare, acestea vor fi ușor de adăugat, fără a intra în conflict cu caracteristicile existente.

* **Scalabilitate**

Aplicația este construită în așa măsură încât, în cazul în care numărul utilizatorilor și al efectuării de sarcini concurente va crește, sistemul va fi capabil să răspundă prompt, fără ca timpul de răspuns să fie alterat în niciun fel.

* **Disponibilitate**

Întrucât aplicația va consta într-un site web, acesta va fi disponibil 24 de ore pe zi, indiferent de tipul de utilizator care dorește să îl acceseze.

* **Utilizabilitate**

Modulele aplicației sunt ușor de utilizat pentru fiecare persoană care se loghează în aplicație. Mai mult, design-ul aplicației va fi plăcută și simplă, indiferent de platforma folosita.

* **Testabilitate**

Scenariile din cadrul aplicațiilor sunt clare și ușor de urmărit, în cazul în care vor fi efectuate teste.

Totadă, dezvoltarea proiectului a implicat și existența unui număr de cerințe nonfuncționale conexe dezvoltării, acestea fiind descrise în continuarea aceste secțiuni.

* Implementarea proiectului folosind limbajul Java pentru back-end și framework-ul React (React.js) pentru front-end
* Integrarea cu Google Authenticator pentru generarea unui cod care va ajuta utilizatorii să se autentifice cu 2FA (engl. *2-factor authentication*)
* Implementarea aplicației astfel încât să poată fi utilizată de pe orice dispozitiv (tabletă sau telefon), indiferent de dimensiunea ecranului
* Testarea manuală a aplicației pentru a determina dacă toate componentele funcționează corect
* Adăugarea comentariilor în codul aplicației

## Motivație

Motivația din spatele creării acestui proiect constă din faptul că majoritatea aplicațiilor din câmpul medical sunt, de cele mai multe ori, unidirecționale, fiind folosite fie doar de pacienți, fie doar de personalul medical.

Aplicația dezvoltată dorește să ofere posibilitatea eficientizării sarcinilor și proceselor care intervin în cadrul unui cabinet medical, oferind functionalități atât angajaților cabinetului, cât și pacienților acestora, în vederea creșterii gradului de transparență și a economisirii de timp pentru ambele părți. Totodată, sarcinile auxiliare (crearea de noi utilizatori sau editarea datelor pentru utilizatorii existenți – referire la angajații unui cabinet) necesită un timp redus pentru a fi executa, dată fiind existența rolurilor administrative.

# Studiu Bibliografic

## Introducere

În prezent, tehnologia este integrată în viețile noastre într-un procent însemnat, fiind capabilă să ne ofere soluții simple și eficiente pentru a rezolva diferite probleme pe parcursul unei zile. Prezența tehnologiei în viețile noastre ne eficientizează procesele în care suntem implicați, fapt ce se referă și la activitatea noastră la locul de muncă; prin natura sa, *un mediu de muncă digitalizat* oferă oportunități de a-ți desfășura activitatea într-un mod mai *eficient*, *crescând productivitatea* *și nivelul de implicare al angajaților* [5]. Cu toate că există și provocări în digitalizarea locului de muncă, acest proces este, de cele mai multe ori, *fundația pentru o strategie de business de success* [5].

## Situația aplicațiilor în domeniul medical

Un studiu [4] realizat în 2014 susține că aplicațiile din domeniul medical sunt *de nișă*, acest lucru urmând *să se schimbe*, putem spune că în prezent, 6 ani mai târziu, aceste tipuri de aplicații sunt din ce în ce mai folosite, mai variate și mai complexe. Deși numărul accestor tipuri de aplicații a părut să stagneze în jurul valorii de *100 000* în 2014 [4] și 2015 [3], numărul acestora este într-o continuă creștere, atingând valoarea de *peste 156 000* în 2016 [6].

Digitalizarea este prezentă în toate câmpurile vieții noastre, permițându-ne o varietate de funcționalități noi și mai complexe, susținând astfel inovarea proceselor în care suntem implicați până la momentul actual. *Revoluția telecomunicațiilor* [7] a ajutat în obținerea de progrese semnificative în câmpul medicinei, astfel că *telemedicina* poate fi văzută drept *rezultatul direct al aplicării acestei revoluții*. În prezent, s-a constatat că *tele-operația robotică de la distanță* (engl. *Remote Robotic Telesurgery*) este *fezabilă, lipsită de riscuri și utilă*, fapt ce reprezintă un progres semnificativ. Totodată, consultațiile la distanță (engl. *Teleconsultation*) și mentorarea la distanță (engl. *Telementoring*) au *fost aplicate din ce în ce mai mult* [7]*.*

**TODO: Extragere informații**

**How smartphones are changing the face of mobile and participatory healthcare: an overview, with an example from eCAALYX** [8]

**TODO: Extragere informații**

**Applications of Computer Simulation in Medical Scheduling** [9]

**TODO: Extragere informații**

**eHealth Applications in Health Care Management** [10]

## Tipurile de aplicații din domeniul medical

**TODO: Extragere informații din:**

[11][12][13][14]

**DRAFT:**

**Varianta aplicatiei WEB**

Prima dintre variante presupune faptul ca aplicatia va putea fi accesata dintr-un browser, adaptandu-se faptului ca este folosita pe un smartphone sau pe o tableta. Aceste tipuri de aplicatii necesita un timp de implementare redus, ceea ce se traduce implicit printr-un cost mai redus. Cu toate acestea, odata cu acest avantaj, se introduce si un dezavantaj, reprezentat de faptul ca aceste aplicatii nu au o interactiune cu utilizatorul la fel de inalta calitativ precum o aplicatie nativa. Totodata, anumite controale specifice unei aplicatii native – accesarea locatiei, a fisierelor de memorie – sau folosirea anumitor senzori – acceleromentru – nu este la fel de facila. Varianta de implementare descrisa anterior este folosita cu precadere cand aplicatiei nu ii sunt necesare conexiunile cu senzori si functionalitatile oferite de dispozitivul de pe care este lansata aplicatia. Principalul avantaj al acestei optiuni il reprezinta faptul ca resursele necesare dezoltarii sunt reduse si ca aplicatia va fi adaptabila oricarui tip de ecran (calculator, tableta sau smartphone), design-ul acesteia fiind conceput astfel incat sa se poata mula oricarei rezolutii.

**Varianta aplicatiei NATIVE**

Cea de-a doua varianta presupune implementarea unei aplicatii native, care poate fi implementata pentru una sau mai multa platforme – Android, iOs sau Windows Phone. Aceste tipuri de aplicatii sunt cel mai des folosite in momentul in care functionalitatile necesita access la senzorii si functiile pe care le ofera dispozitivu - accesul la camera foto si la sistemul de fisiere, accesul la locatie, acceleromentru, etc – aspecte pe care o aplicatie migrata catre mobil nu le poate gestiona la fel de usor. Cu toate acestea, implementarea unei astfel de aplicatii are nevoie de mai multe resurse (timp si costuri), iar pentru ca aplicatia sa poata fi folosita pe mai multe platforme, trebuie sa existente o implementare proprie pentru fiecare dintre ele. Totodata, multitudinea de versiuni pentru sistemele de operare si dispozitivele pe care se va utiliza aplicatia va creste considerabil timpul de dezvoltare.

Luand in considerare atat avantajele si dezavantajele oferite de celel doua moduri de realizarea a aplicatiilor mobile, am ajuns la concluzia ca, pentru acest proiect, alegerea primei variante – aplicatie mobile web – este cea care ar putea sa fie cea mai potrivita, luand in calcul faptul ca aplicatia web va fi cea dezvoltata prima, reprezentand o baza de pornire pentru dezvoltarea pe mobile (migrare).

## Studiul aplicațiilor existente pentru cabinetele medicale

În aria aplicațiilor folosite în domeniul medical, cu precădere pentru cabinete, există mai multe aplicații oferite de diverși dezvoltatori, dintre care vom analiza patru aplicații: icMed, Medsoft, Evolvo și MedXLine.

### icMed

*icMed* [15] este un sistem informatic medical online, sub forma unei aplicații web, dezvoltat de compania Syonic și introdus în anul 2007, acest sistem având la bază fișa unică a pacientului (EHR – engl. *Electronic Health Record*).

Principalele caracteristici ale aplicației pentru angajații cabinetului sunt:

* Managementul pacienților prin EHR
* Managementul diagnosticelor
* Generarea de rapoarte
* Eliberarea de rețete
* Crearea de consultații și completarea fișei pacientului
* Atașarea de tratamente
* Completarea fișei de examen clinic



Figura 3.1 Simbolul *icMed* (Sursă: [15])

Există și o aplicație pentru pacienți, *icMED.Mobile* [16], aceasta fiind disponibilă sub forma unei aplicații mobile native (pentru Adroid și iOS). Pacientul primește acces odată ce doctorul îi trimite acestuia o invitație sub forma unui mesaj text, conținând datele de autentificare necesare.

Caracteristicile aplicației mobile pentru pacienții cabinetului sunt:

* Solicitarea de programări
* Vizualizarea rețetelor prescrise
* Schimbare de mesaje, documente sau conținut media cu medicii
* Vizualizarea rezultatelor la analize sau imagistică (dacă analizele sunt efectuate în rețeaua icMED)



Figura 3.2 Simbolul *icMED.Mobile* (Sursă: [16])

### Medsoft

*Medsoft* [17], produs de compania cu acelați nume, reprezintă un software de gestiune pentru clinicile medicale, o unealtă care poate fi utilizată de personalul cabinetului medical pentru realizarea diverselor sarcini. Aplicația, sub forma unui site web, este dezvoltată atât pentru utilizarea de pe un calculator, cât și de pe tabletă sau de pe un smartphone.

Principalele caracteristici oferite de această aplicații sunt:

* Managementul programărilor pacienților
* Gestionarea unui dosar electornic pentru fiecare pacient
* Modul de casierie
* Generarea de rapoarte
* Crearea de consultații și completarea fișei pacientului
* Atașarea de tratamente
* Completarea fișei de examen clinic



Figura 3.3 Simbolul *MedSoft* (Sursă: [17])

### Evolvo

Aplicația Evolvo [18] se prezintă sub forma unui software online creat pentru medici, care poate fi accesat de pe un calculator, folosind un browser web, fiind capabilă, dacă este accesată în acest mod, să ofere întregul spectru de funcționalități disponibile.

Funcționalitățile oferite de aplicația web sunt:

* Gestiunea pacienților și crearea de programări pentru aceștia
* Existența unei fișe medicale online
* Gestiunea tratamentelor
* Modul pentru date financiare
* Modul pentru date de marketing
* Acces la date de imagistică
* Crearea și vizualizarea de rapoarte
* Suport pentru educarea pacienților
* Posibilitatea semnării digitale

Totodată, există și o aplicație nativă (Android [19] sau iOS [20]), care poate fi accesată de pe un telefon sau de pe o tabletă, acesta reprezentând un instrument auxiliar, pe care doctorii îl pot folosi în lipsa accesului la un calculator, de pe care să poată acceasa aplicația web.

Funcționalitățile oferite de aplicația mobilă sunt:

* Gestiunea pacienților și a programărilor
* Acces la date de imagistică



Figura 3.4 Simbolul *Evolvo* (Sursă: [18])

### MedXLine

Aplicața MedXLine [21], dezvoltată de către compania LifeIsHard reprezintă o suită de module de tip aplicație, fiecare dintre acestea oferind funcționalitate fie în sfera proceselor operaționale sau administrative, fie în cea a proceselor de natură strict medicală.

Secționarea aplicației pe module este concepută pentru a se plia nevoilor oricărei fel de unitate medicală, insituțiile medicale fiind capabile să își selecteze modulele pe care le vor utiliza.

Modulele disponibile ale MedXLine sunt:

* Gestiunea pacienților (exită un modul mai specific, conținând și funcționalitatea de *recepție pacienți*, dezvoltat pentru instituțiile de tip *clinică*)
* Efectuarea de consultații
* Gestiunea învestigațiilor (radiologie și oftalmologie)
* Gesitiunea analizelor medicale
* Generarea de rapoarte
* Gestiunea stocului de mateiale medicale
* Gestionarea contractelor și abonamentelor pentru clienții unității medicale



Figura 3.5 Simbolul *MedXLine* (Sursă: [21])

# Analiză şi Fundamentare Teoretică

## Soluția propusă

În cadrul aplicației, toți angajații (doctori și asistenți medicali deopotrivă) și pacienții sunt înregistrați în cadrul cabinetului din care fac parte, fiind capabili să acceseze întregul spectru de funcționalități specifice rolului pe care îl dețin în cadrul cabinetului. Aplicatia suportă și existența rolurilor administrative, astfel că există două astfel de roluri: administratorul de aplicație și administratorul de cabinet. Administratorul de aplicație poate fi văzut ca un superadministrator, fiind singurul rol cu funcționalitatea de a înregistra un nou cabinet și de a edita datele tuturor cabintelor din sistem, cât și a tuturor utilizatorilor (administratori de cabinete și personal medical). Având parte de o funcționalitate similară, administratorul de cabinet poate efectua modifiări asupra datelor, cu precizarea că acest rol poate edita doar datele propriului cabinet și al angajaților care țin de acesta.

## Motivarea soluției

Alegerea tipului de aplicație a fost o decizie informată, deoarece au fost luate în calcul atât caracteristicile aplicațiilor native (pentru Android sau iOS), cât și ale unei aplicații web portabile, scopul acestei analize fiind determinarea tipului de aplicație care se pliază cel mai bine pe tipul de proiect de realizat. Pentru mai multe detalii ale procesului de analiză, se poate consulta capitolului anterior al acestei lucrări.

În urma analizei, s-a ales ca soluția dezvoltată să se prezinte sub forma unei aplicații web, care este disponibilă atât de pe un calculator, cât și de pe o tabletă sau de pe un smarphone. Tehnologiile folosite pentru realizare frontend-ului aplicației (versiunea 5 a limbajului HTML și a framework-ului Bootstrap de CSS) au servit drept instrumente importante pentru realizarea interfeței capabile să confere portabilitatea aplicației pe mai multe dispozitive, acestea având diferite rezoluții.

unde trebuie relevata utilitatea

aplicatiei din punctul de vedere al potentialilor beneficiari (cabinete

medicale, respectiv pacienti).

## Funcționalitatea și rolurile aplicației

NU ar strica sa implementezi in aplicatie si posibilitatea ca pacientii

sa poata furniza de acasa parametri pe care si-i masoara singuri, precum

hipertensiunea arteriala, temperatura, pulsul, eventual glicemia, precum

si simptomatologia care ii ingrijoreaza, in acest mod putand fi

consultati si in regim de urgenta, prin intermediul telemedicinei (de la

distanta), iar medicul sa le poata raspunde in mod text, prin oferirea

de indicatii si stabilirea unui diagnostic.

Aplicația dezvoltată oferă funcționalitate pentru un număr de 5 roluri bine-delimitate. Aceste roluri sunt: pacient, cadre medicale ale cabinetului (asistent medical și doctor), cât și uadministratori (de cabinet și de aplicație). Fiecare dintre aceste roluri are acces la funcționalități specifice.

Pentru o viziune în ansamblu asupra rolurilor și funcționalităților asociate, se pot consulta următoarele diagrame ale cazurilor de utilizare, acesta urmând să fie reluate în cadrul următorului capitol pentru o analiză mai complexă.

Rolul de doctor are un număr de 9 funcționalități, acestea fiind atât de natură administrativă (gestionarea programărilor primite de la pacienți), cât și de natură medicală (gestionarea programărilor acceptate – a consultațiilor – prin adăugarea detaliilor medicale necesare).

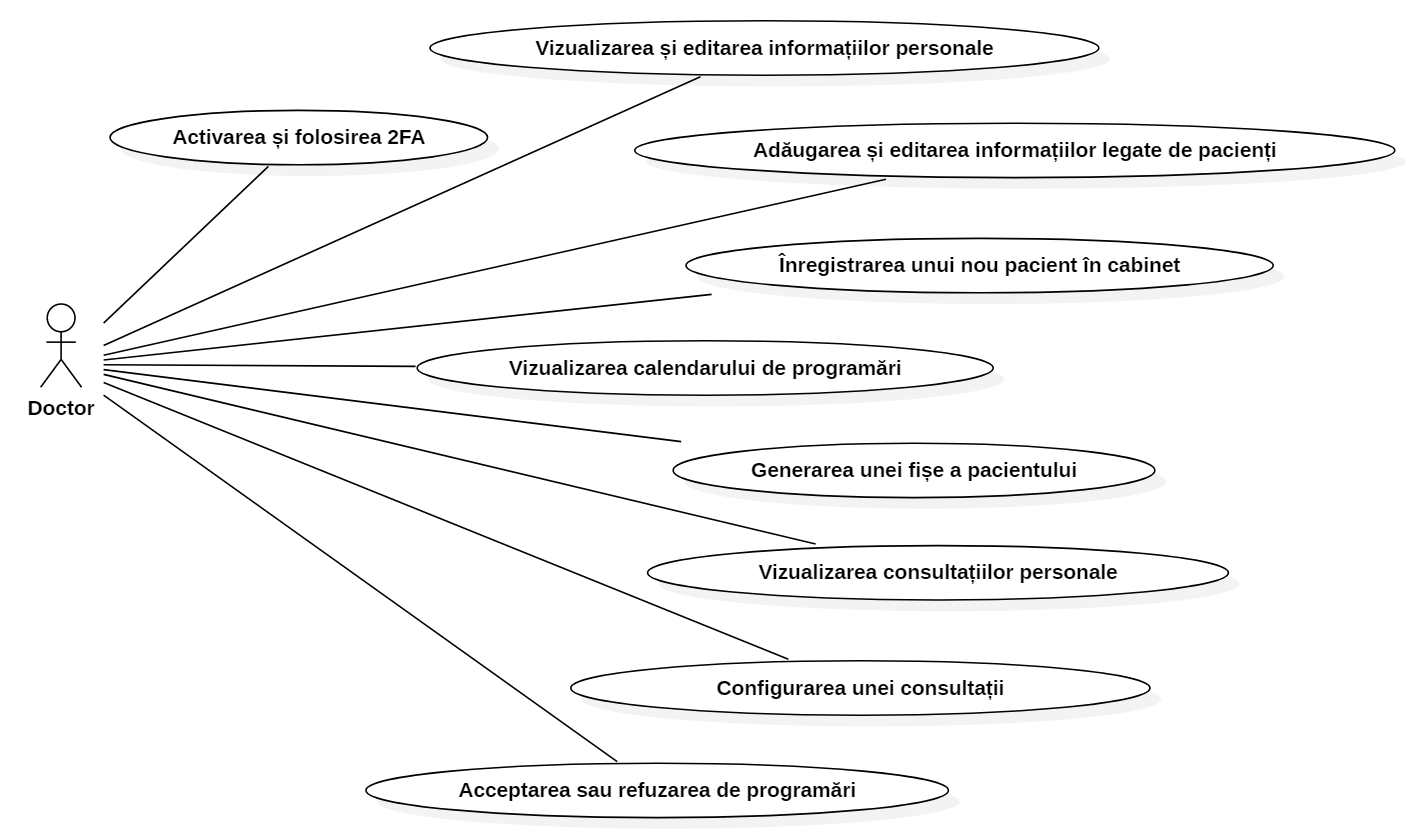


Figura 4.1 Diagrama cazurilor de utilizare pentru rolul de doctor

Rolul de asistent medical deține un număr de 7 funcționalități, toate acestea având un rol administrativ, atât legat de pacienții cabinetului, cât și de programările făcute către medicii acestuia, astfel că asistenții medicali pot să gestioneze programările pacienților, dar și entitățile corespondente acestora.

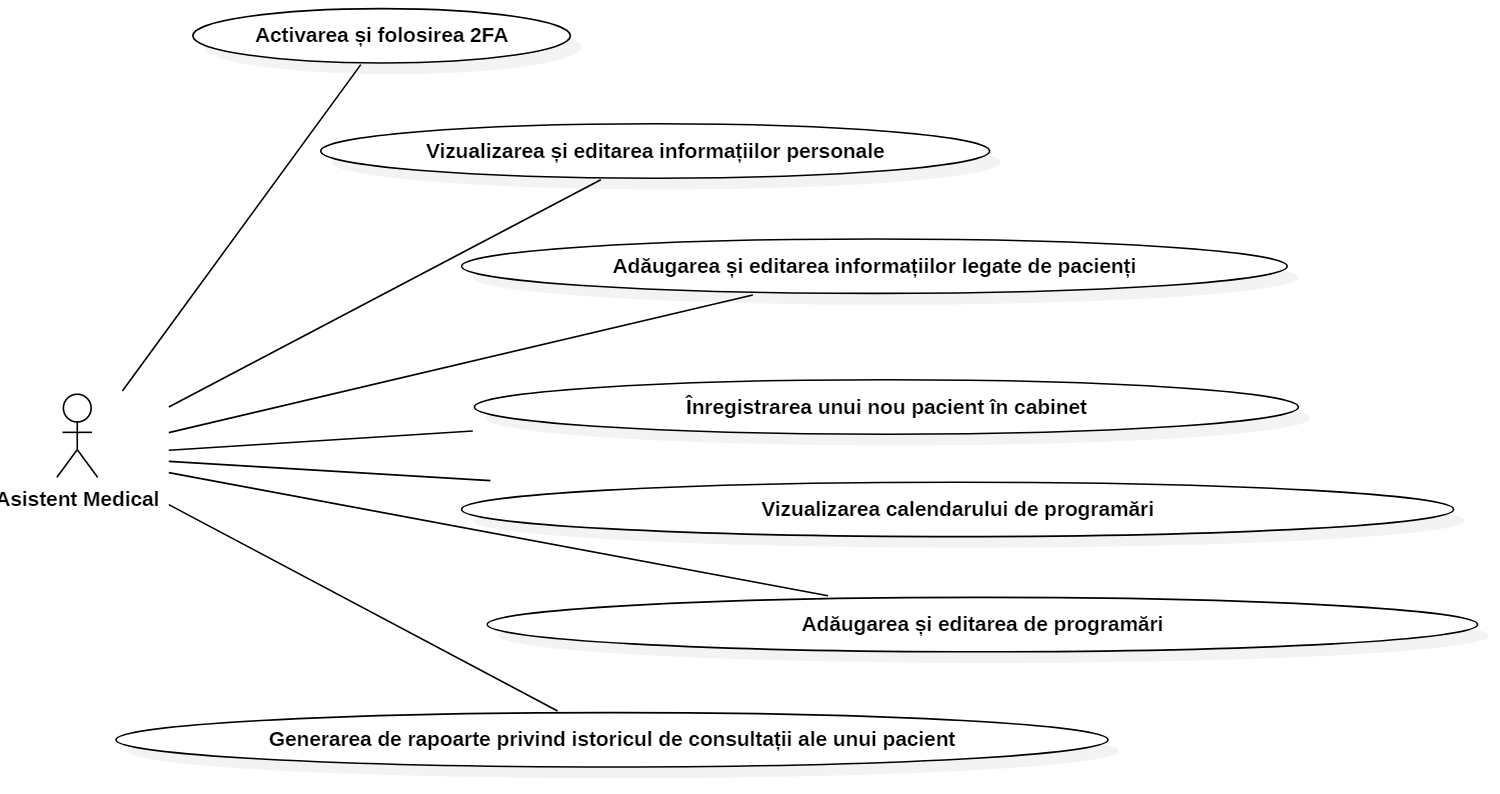


Figura 4.1 Diagrama cazurilor de utilizare pentru rolul de asistent medical

Rolul de pacient aduce cu sine un număr de cinci funcționalități, cu ajutorul cărora pacientul poate să interacționeze cu cabinetul medical, fiind capabil să solicite programări, să își actualizeze informațiile personale, dar și pe cele de natură medicală.

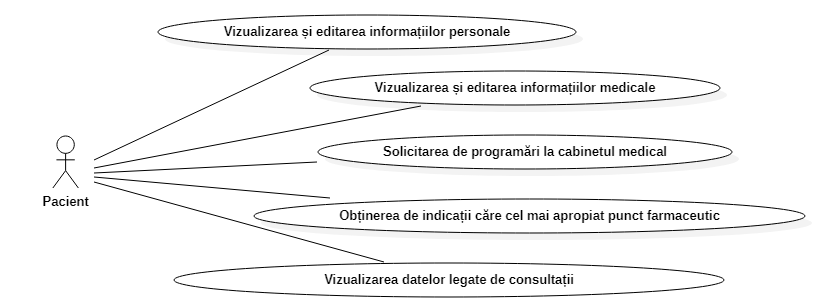


Figura 4.3 Diagrama cazurilor de utilizare pentru rolul de pacient

Rolul de administrator de cabinet aduce cu sine funcționalități în strictă legătura cu datele specifice cabinetului pe care acesta îl administrează, precum gestionarea informațiilor legate de angajații cabinetului, cât și editarea informațiilor legate de entitatea cabinetului.

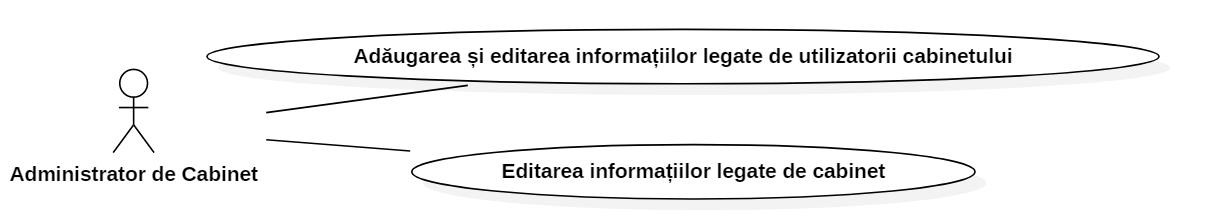


Figura 4.2 Diagrama cazurilor de utilizare pentru rolul de administrator de cabinet

Rolul de administrator de aplicație are atribuite 2 funcționalități, ambele conferind control absolut asupra datelor legate de angajații tuturor cabinetelor (asistenți și doctori deopotrivă), cât și asupra cabinetelor în sine, acest tip de utilizator fiind capabil să adauge noi entități, să le editeze datele sau să le șteargă.

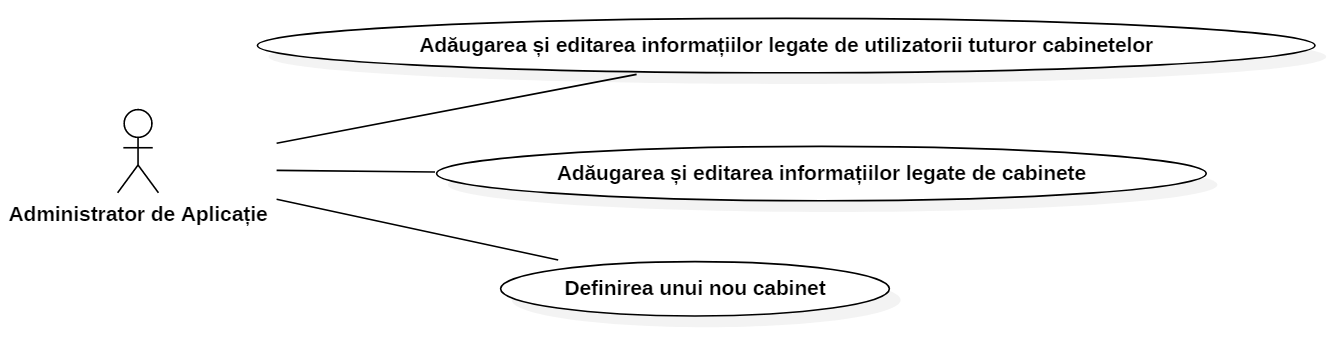


Figura 4.3 Diagrama cazurilor de utilizare pentru rolul de administrator de aplicație

## Tehnologiile și uneltele alese pentru dezvoltarea aplicației

### Java Spring

Limbajul folosit pentru dezvoltarea părții de back-end a fost Java, iar framework-ul folosit a fost Spring. Utilizarea acestui limbaj, care face parte din paradigma orientată pe obiecte, conferă un grad ridicat de eficiență în vederea dezvoltării unei arhitecturi robuste și sigure. Legat de alegerea framework-ului Spring, un factor major în această decizie a fost influențată de numărul semnificativ pe care acest framework le oferă [22]. Cele cu o însemnătate ridicată în dezvoltarea acestei aplicații au fost faptul că nu este nevoie de *un server web sau software pentru serverul de aplicații* pentru a lansa aplicația. Totodată, *Spring MVC* (Model-View-Controller)este capabil să *ofere un cadru web puternic și flexibil* și, nu în ultimul rând, Spring oferă suport pentru *framework-ul JDBC* (Java DataBase Connectivity), ajutând la *reducerea erorilor*.



Figura 4.4 Simbolul Java Spring (Sursă: [23])

### HTML5, Bootstrap și React

Pentru dezvoltarea părții de front-end a aplicației, am ales să folosesc versiunea cu numărul 5 a HTML, framework-ul Bootstrap pentru CSS și framework-ul React pentru partea de JavaScript.

De adăugat: Mai multe detalii

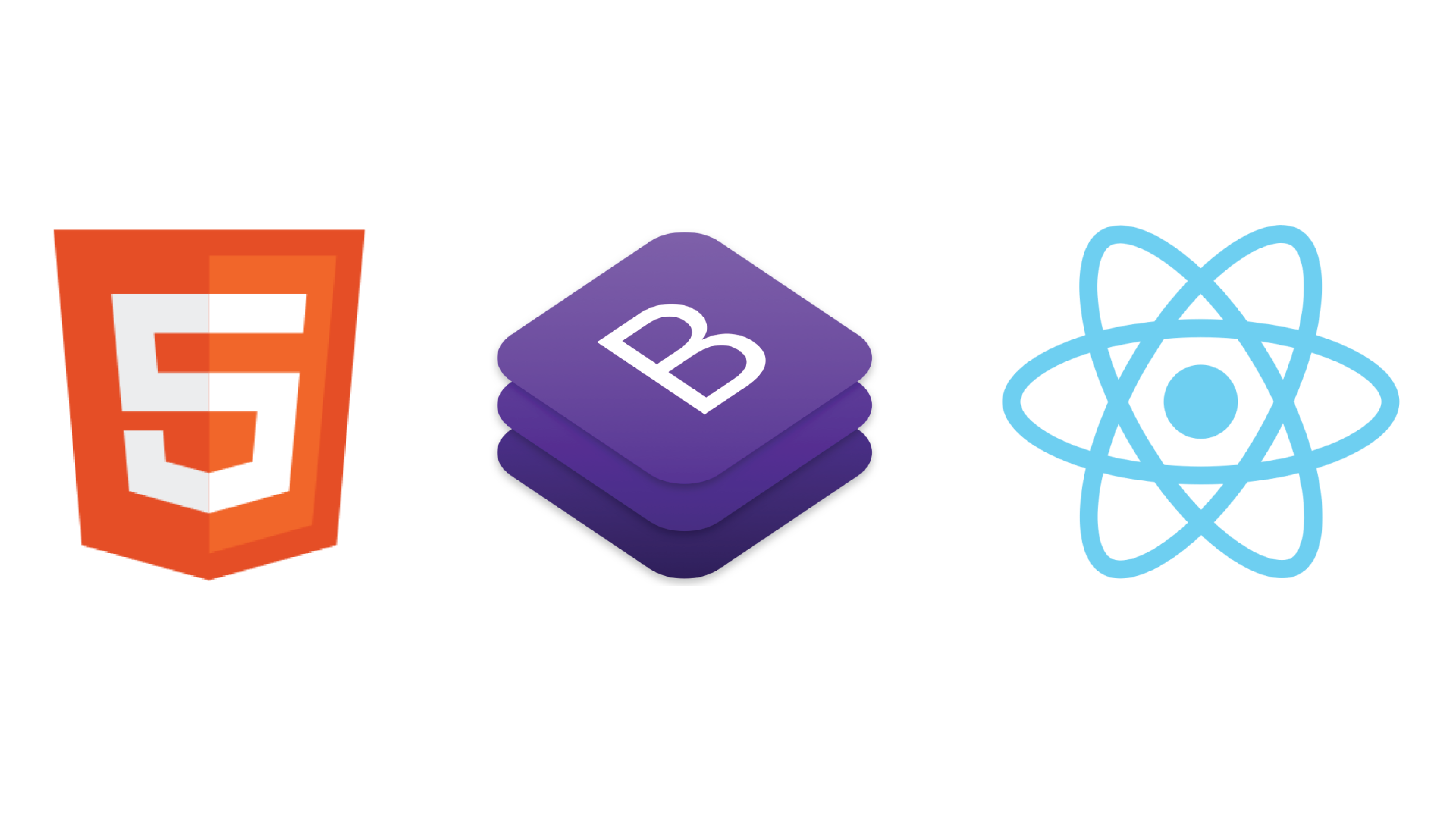


Figura 4.5 Simbolurile HTML5, Bootstrap și React (Surse: [24], [25], [26])

## Dependețe externe

### Google Authenticator

În cadrul acestei aplicații este folosită o singură dependință externă, aceasta fiind o aplicație creată de către Google, care oferă *funcționalitatea de 2FA*(engl. *Two-Factor Authentication*)[27]. Acest sistem se folosesțe de mecanismul *TOTP* (engl. *Time-based one-time password*) pentru *generarea unei secvențe de coduri de verificare*, acesta fiind folosită în momentul în care un utilizator se autentifică.

De adăugat: Mai multe detalii



Figura 4.6 Simbolul Google Authenticator(Sursă: [9])

# Proiectare de Detaliu si Implementare

## Specificațiile sistemului

Sistemul se prezintă sub forma unei aplicații web care poate fi folosită în browser, atât de pe un calculator, cât și de pe un telefon, interfața acesteia adaptându-se la dimensiunea ecranului dispozitivului, fiind capabilă să ofere aceeași experiență, indiferent de specificațiile acestuia.

## Arhitectura sistemului

## Cazurile de utilizare

În cadrul acestei secțiuni se vor analiza cazurile de utilizare care există în cadrul aplicației. Acestea vor fi abordate pentru fiecare dintre tipurile de roluri disponibile (pacienți, cadre medicale și administratori), fiecare caz fiind analizat folosind structura următoare: actorul, diagrama acțiunilor, pașii scenariului, scenarii alternative, cerințe speciale, precondiții și postcondiții, cât și puncte de extensie, unde acest aspect este fezabil.

Pentru toate cazurile de utilizare este nevoie să fie îndeplinite două precondiții care, din cauză ca ar fi reduntant să fie amintite pentru fiecare caz de utilizare în parte, vor fi explicate în continuare iar, pentru cazurile care necesită și alte precondiții, acestea vor fi specificate corespunzător.

1. Existența unei conexiuni stabile la internet

Această conditie se vede necesară din moment ce aplicația rulează într-un tab de browser și, totodată, pentru a evita situațiile în care datele nu se pot salva din cauza lipsei de stabilitate a conexiunii.

1. O autentificare prealabilă în aplicație

Pentru a putea accesa această funcționalitate este nevoie ca actorul să fie logat drept un pacient, acesta fiind singurul rol care poate să efectueze acțiunea de a solicita programări.

Un alt element comun al tuturor cazurilor de utilizare sunt și trei scenarii alternative care vor fi enunțate în continuare. Următoarele două scenarii sunt menționate la momentul actual pentru a fi amintite, în mod redundant, pentru fiecare dintre scenarii. În cazul în care există și alte scenarii alternative, acestea vor fi menționate pentru fiecare caz în parte.

1. Abandonarea procesului de a executa sau confirma o acțiune

Actorul poate să se oprească din executarea unui scenariu în orice punct, putând să închidă tab-ul din browser în care aplicația rulează. În cazul în care nu există modificări ale datelor, această acțiune poate fi efectuată fără nicio avertizare de la aplicație. În cazul în care există modificări și actorul refuză să le salveze înainte să închidă aplicația, modificările efectuate se vor pierde, iar baza de date își va păstra statutul inițial pentru ca datele să rămână valide, evitând astfel conflicte și erori ulterioare în aplicație.

1. Salvarea sau confirmarea unor datelor cauzează apariția unei erori

În cazul în care actorul introduce date invalide (cifre sau caractere speciale în câmpuri care nu suportă un astfel de input), aplicația va afișa mesaje de eroare pentru a îndruma utilizatorul spre a-și actualiza datele introduse într-un mod corespunzător.

1. Apariția unei erori neașteptate

Acest scenariul este posibil în oricare dintre pașii scenariului. În cazul în care actualizarea datelor nu este posibilă, modalitate prin care actorul și le-ar putea utiliza ar fi prin contactarea unui utilizator cu rol de administrator, care are disponibile operații de editare a datelor pentru angajații unui cabinet.

### Cazurile de utilizare ale angajaților cabinetului

Actorii următoarelor cazuri de utilizare sunt fie asistentenții medicale, fie doctorii. Cele două roluri au atât funcționalități comune, cât și unele specifice. În continuare, se vor trata scenariile comune, urmate de cele specifice unui anumit rol.

#### Cazurile de utilizare comune asistenților medicali și doctorilor

Există un număr de patru cazuri de utilizare ale căror actor poate fi atât un utilizator care are rolul de asistent medical, cât și de doctor, acestea fiind prezentate în continuare.

##### Activarea și autentificarea folosind 2FA

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată profilului personal
2. Actorul alege opțiunea ”Autentificare folosind 2FA”
3. Actorul părăsește pagina
   1. Dacă modificările nu sunt salvate, modalitate de autentificare nu va fi schimbată, urmând ca actorului să îi fie cerută doar parola la următoarea autentificare în aplicație.
   2. Dacă modificările sunt salvate, actorului îi rămâne salvată opțiunea de a se autentifica folosind parola și un cod unic, generat de aplicația Google Authenticator
4. Actorul se deconectează din aplicație
5. Actorul își introduce datele primite prin email în Google Authenticator pentru a-și primi codul pe care îl va utiliza la autentificarea în aplicație
6. Actorul se autentifică folosindu-și parola și codul din Google Authenticator

**Scenarii alternative:**

1. Email-ul conținând datele pentru configurarea 2FA nu este primit de către actor

În cazul în care actorul nu primește un email cu datele pe care le poate folosi pentru a-și configura autentificarea cu 2FA, acesta este nevoit să refacă procesul de logare, iar, dacă acest lucru nu este posibil, va trebui să anunțe administratorii aplicației îm legătură cu acest incident.

**Cerințe speciale:**

1. Pentru acest caz de utilizare există o singură cerință, aceasta fiind că utilizatorul trebuie să aibă aplicația Google Authenticator instalată pe un smartphone sau pe o tabletă. Instrucțiunile pentru configurarea 2FA sunt transmise printr-un email, utilizatorul urmând să își introducă datele în aplicația Google Authenticator pentru a-și genera codul de autentificare unic.

**Postcondiții:**

1. Autentificarea se face atât cu o parolă, cât și un cod unic

Atunci când utilizatorul se autentifică, acestuia îi este cerută atât parola contului, cât și un cod unic, care poate este generat și poate fi preluat din aplicația Google Authenticator.

1. Utilizatorul este capabil să completeze procesul de autentificare cu success

Odată ce procesul de activare al autentificării de tipul 2FA a fost completat cu success, utilizatorii se pot autentifica în aplicație, fiind capabilă să întreprindă diferite acțiuni.

**Diagrama de evenimente:**

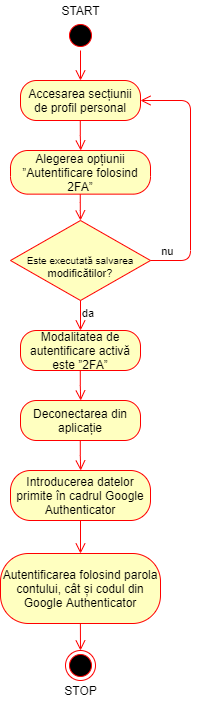


Figura 5.1 Diagrama de evenimente a cazului de utilizare ”Activarea și autentificarea folosind 2FA”

##### Vizualizarea și editarea informațiilor personale

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată profilului personal
2. Actorul își vizualizează datele personale
   1. Dacă actorul alege să nu execute modificări, atunci scenariul ajunge la final
   2. Dacă actorul alege să își modifice datele, atunci scenariul continuă
3. Actorul este în pasul de confirmare a modificărilor făcute
   1. Dacă noile date salvate, modificările făcute se vor pierde
   2. Dacă actorul alege să își salveze modificările, datele acestuia vor fi actualizate cu noile valori

**Postcondiții:**

1. Odată ce datele sunt salvate cu success, actorul va fi capabil să își vizualizeze datele, acestea având ultimele valori salvate de către acesta.

**Diagrama de evenimente:**

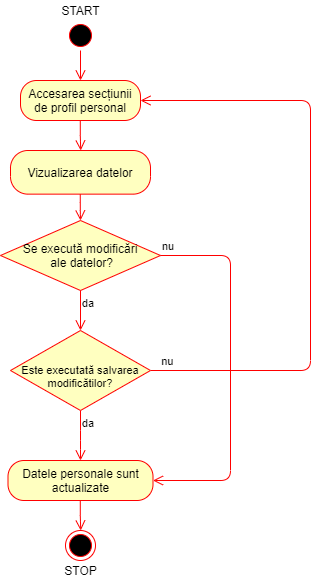


Figura 5.2 Diagrama de evenimente a cazului de utilizare ”Vizualizarea și editarea informațiilor personale”

##### Adăugarea și editarea informațiilor legate de pacienți

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată datelor despre pacienți
2. Actorul alege să vizualizeze datele unui pacient specific
   1. Dacă actorul nu dorește să modifice datele pacientului, scenariul se sfârșește după această acțiune
   2. Dacă actorul dorește să modifice datele pacientului, scenariul continuă
3. Actorul este în pasul de confirmare a modificărilor făcute (dacă există)
   1. Dacă actorul alege să nu salveze modificările făcute și părăsește pagina, datele introduse de acesta nu se vor salva, pacientul având aceleași date ca și la începerea acestui scenariu
   2. Dacă actorul alege să salveze modificările făcute pentru datele pacientului, profilul al pacientului se va actualiza

**Postcondiții:**

1. Odată ce actorul modifică datele pacientului și alege să își salveze modificările, acestea vor fi vizibile atât pentru angajații cabinetului, cât și pentru pacient.

**Diagrama de evenimente:**

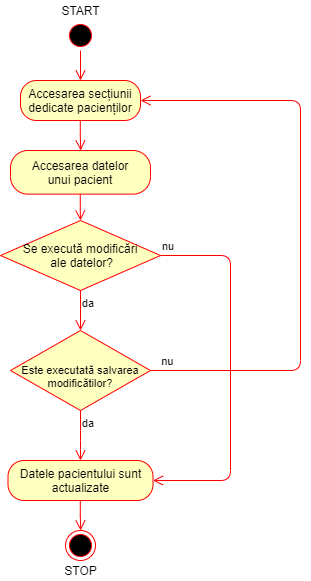


Figura 5.3 Diagrama de evenimente ”Adăugarea și editarea informațiilor legate de pacienți”

##### Înregistrarea unui nou pacient în cabinet

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată pacienților
2. Actorul optează pentru înregistrarea unui pacient nou
3. Actorul începe acțiunea de a introduce datele pacientului
   1. Dacă datele introduse nu sunt valide, câmpurile care conțin astfel de date vor fi semnalate corespunzător.
   2. Dacă datele sunt valide, scenariul poate continua
4. Actorul este în pasul de confirmare a acțiunii de înregistrare a unui nou pacient
   1. Dacă date noului pacient nu sunt salvate, modificările făcute se vor pierde
   2. Dacă actorul alege să își salveze modificările, un nou cont de pacient va fi creat, iar datele acestuia cont vor fi cele introduse anterior de actor

**Postcondiții:**

1. După ce actorul introduce date valide, confirmă acțiunea de a înregistra un nou pacient și este notificat de sistem că acțiunea a fost executată cu success, pe email-ul pacientului se vor primi datele de autentificare în aplicație.
2. Noul pacient va apărea în secțiunea destinată pacienților, atât pentru rolul de asistent medical, cât și de doctor.

**Puncte de extensie ale scenarului:**

1. Având un cont creat, pacientul poate să acceseze aplicația și, având acest aspect drept punct de pornire, acesta are posibilitatea să-și adauge și editeze informațiile personale, să creeze solicitări pentru programări, cât și să primească indicații către cel mai apropiat punct farmaceutic după participarea la o consultație (o programare acceptată devine o consultație)

**Diagrama de evenimente:**

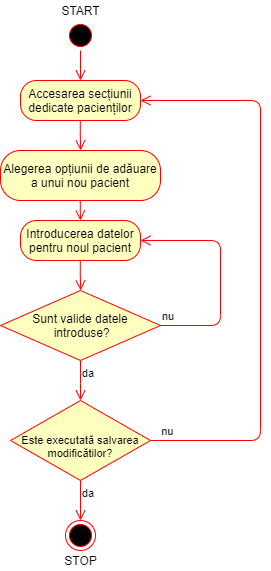


Figura 5.4 Diagrama de evenimnete ”Înregistrarea unui nou pacient în cabinet”

##### Vizualizarea calendarului de programări

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată programărilor
2. Actorul alege să intre pe tab-ul corespondent calendarului de programări. Scenariul ia sfârșit odată cu acest pas.

**Puncte de extensie:**

1. În cazul în care există programări cărora nu li s-a asociat un status (acceptată sau refuzată), actorul poate să aleagă statusul pe care dorește să îl aleagă pentru aceste tipuri de programări.
2. Odată ce unei programări i se asociază un status, aceasta este afișată într-un mod diferit pe calendarul actorului; dacă programarea este acceptată, statusul acesteia nu va mai putea fi schimbat, iar intervalul alocat programării va fi afișat pe calendar, însă, în cazul în care programarea este refuzată, aceasta nu va mai fi vizibilă pe calendarul actorului.

**Diagrama de evenimente:**

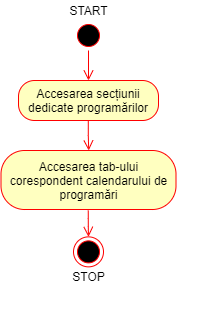


Figura 5.5 Diagrama de evenimente ”Vizualizarea calendarului de programări”

#### Cazurile de utilizare specifice asistenților medicali

Pentrul acest rol există un număr de două cazuri de utilizare specifice, care vor fi prezentate în continuare, actorul acestora fiind un asistent medical.

##### Generarea de rapoarte privind istoricul de consultații ale unui pacient

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată rapoartelor
2. Actorul vizualizează lista pacienților și selectează un anume pacient
3. Actorul optează pentru opțiunea de generare a raportului
   1. Dacă pacientul selectat nu are nicio consultație, sistemul va anunța actorul în legătură cu acest aspect și îi va cere să selecteze un alt pacient pentru a completa această acțiune
   2. Dacă pacientul are cel puțin o consultație, atunci actorul poate continua cu următorul pas al scenariului
4. Actorul își confirmă acțiunea de generare a raportului

**Postcondiții:**

1. Dacă este selectat un pacient care are un istoric de consultații la cabinet, iar actorul confirmă acțiunea de a genera un raport, atunci un fișier cu extensia .pdf va începe să se descarce automat, acesta conținând întregul istoric de consultații al pacientului.

**Diagrama de evenimente:**

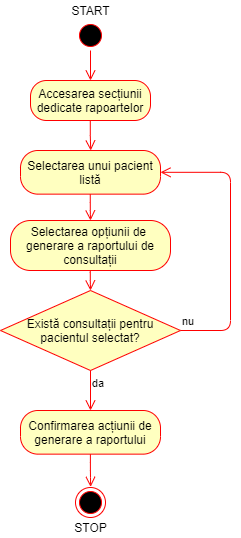


Figura 5.6 Diagrama de evenimente ”Generarea de rapoarte privind istoricul de consultații ale unui pacient”

##### Adăugarea și editarea de programări

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată programărilor
   1. Dacă actorul alege să creeze o nouă programare, atunci următorul pas al scenariul se va regăsi la numărul 3
   2. Dacă actorul alege să vizualizeze (și să posibil modifice) datele unei programări, atunci acesta va continua scenariul cu pasul numărul 2
2. Actorul alege să vizualizeze o programare din listă
   1. Dacă actorul nu dorește să modifice datele programării pe care a ales să o vizualizeze, atunci scenariul se sfârșește după această acțiune
   2. Dacă actorul dorește să modifice datele programării, scenariul continuă cu pasul 3
3. Actorul este în pasul de confirmare a modificărilor făcute
   1. Dacă actorul alege să nu salveze modificările făcute și părăsește pagina, datele introduse de acesta nu se vor salva, programarea având aceleași date ca și la începerea acestui scenariu sau, în cazul creării unei programări noi, această entitate nu va putea fi salvată
   2. Dacă actorul alege să salveze modificările făcute, datele unei programări existente se vor actualiza, sau, în cazul creării uneia noi, aceasta va fi salvată

**Scenarii alternative:**

1. Există încă un rol, pe lângă cel al actorului (asistent medical), care este capabil de crearea programărilor la cabinetul medical: pacientul. Programările solicitate de pacienți apar în lista pe care actorul acestui scenariu poate să realizeze modificări.

**Postcondiții:**

1. Odată ce actorul modifică datele pacientului și alege să își salveze modificările, acestea vor fi vizibile atât pentru angajații cabinetului, cât și pentru pacient.

**Puncte de extensie:**

1. Din moment ce o nouă programare este creată sau datele unei programări existente sunt modificate, aceste schimbări vor fi vizibile, atât pentru pacient, cât și pentru doctor.
2. Odată creată, programarea ajunge să fie afișată în calendarul doctorului, care, ulterior, poate să o refuze sau să o accepte, moment în care pacientul este anunțat de acțiunea care a avut loc printr-un email.

**Diagrama de evenimente:**

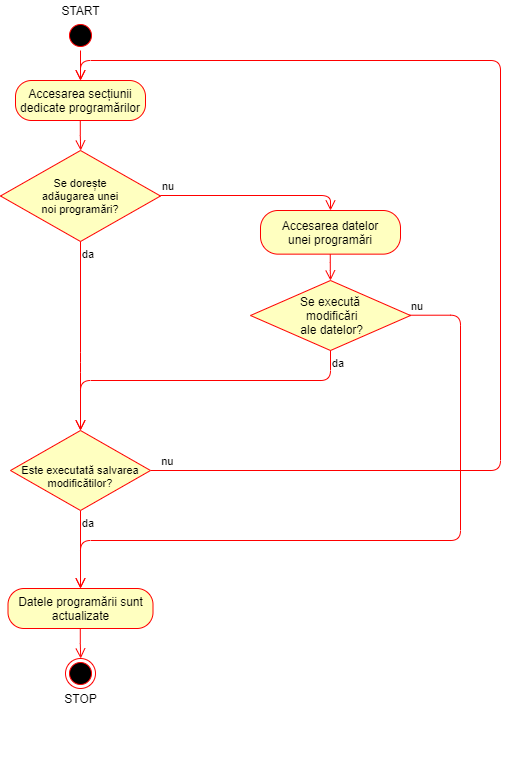


Figura 5.7 Diagrama de evenimente ”Adăugarea și editarea de programări”

#### Cazurile de utilizare specifice doctorilor

##### Generarea unei fișe a pacientului

1. Actorul accesează secțiunea destinată rapoartelor
2. Actorul selectează un pacient din lista afișată
3. Actorul optează pentru generarea fișei pacientului
   1. Dacă actorul nu confirmă această acțiune, nu se va genera niciun fișiser, iar actorul va reveni pe pagina destinată fișelor medicale
   2. Dacă actorul își confirmă acțiunea de a genera fișa pacientului, atunci se va începe descărcarea unui fișier .pdf conținând toate datele medicale ale pacientului

**Postcondiții:**

1. Odată ce generarea fișei este confirmată de actor, acesta ar trebui să poată accesa un fișier .pdf care va începe să se descarce odată cu închiderea dialogului de confirmare a generării.

**Diagrama de evenimente:**

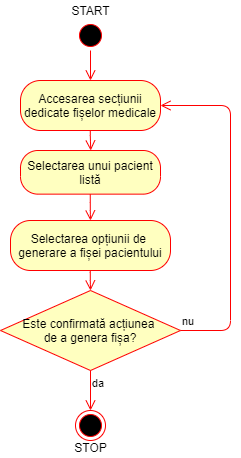
****

Figura 5.8 Diagrama de evenimente ”Generarea unei fișe a pacientului”

##### Acceptarea sau refuzarea de programări

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată programărilorl
2. Actorul alege să intre pe tab-ul corespondent calendarului de programări
   1. Dacă nu există programări al căror status poate fi modificat (prin acceptarea sau refuzarea lor), atunci scenariul ia sfârșit odată cu acest pas.
   2. Dacă există pogramări cărora nu le-a fost asignat un status, atunci actorul continuă scenariul
3. Actorul alege statusul pe care să îl atribuie unei programări
   1. Dacă actorul alege să accepte solicitarea de programare, atunci pacientul va fi informat printr-un email că cererea sa a fost acceptată, iar intervalul de timp corespunzând programării acceptate va fi ”blocat” pe calendarul actorului.
   2. Dacă actorul alege să refuze cererea de programare, atunci pacientul va fi informat printr-un email că cererea sa fost refuzată, iar intervalul de timp care ar fi fost alocat programării nu mai este vizibil pe calendarul actorului.

**Precondiții:**

1. Pentru ca acest scenariu să fie valid, este nevoie ca fie pacientul, fie asistentul medical să fi creat o programare la cabinetul medical.

**Postcondiții:**

1. Odată ce actorul alege să seteze un status programării (să o accepte sau să o refuze), pacientul pentru care era creată programarea va primi un email prin care va fi anunțat în legătură cu decizia luată de actor.
2. Calendarul actorului este actualizat conform acțiunilor pe care acesta le întreprinde

Dacă actorul alege să accepte o programare, aceasta este marcată corespunzător în calendarul acestuia, iar crearea unei alte programări în același interval de timp nu va fi posibilă.

Dacă actorul alege să refuze o programare, această acțiune va rezulta în eliberarea slotului dedicat programării din calendar, acest aspect conferindu-le pacienților să solicite o programare în intervalul respectiv.

**Puncte de extensie ale scenarului:**

1. Odată ce o solicitare de programare este acceptată, acesta devine o consultație și poate fi adresată în secțiunea corespunzătoare a aplicației.

**Diagrama de evenimente:**

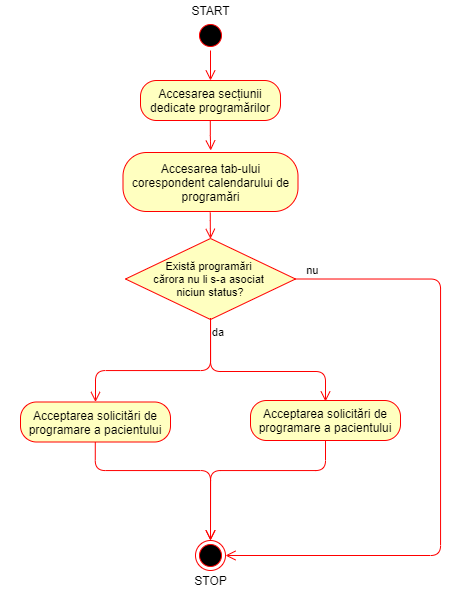


Figura 5.9 Diagrama de evenimente ”Acceptarea sau refuzarea de programări”

##### Vizualizarea consultațiilor atribuite

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată programărilor
2. Actorul alege să intre pe tab-ul corespondent calendarului de programări. Scenariul ia sfârșit odată cu acest pas.

**Puncte de extensie:**

1. În cazul în care există programări cărora nu li s-a asociat un status (acceptată sau refuzată), actorul poate să aleagă statusul pe care dorește să îl aleagă pentru aceste tipuri de programări.
2. Odată ce unei programări i se asociază un status, aceasta este afișată într-un mod diferit pe calendarul actorului; dacă programarea este acceptată, statusul acesteia nu va mai putea fi schimbat, iar intervalul alocat programării va fi afișat pe calendar, însă, în cazul în care programarea este refuzată, aceasta nu va mai fi vizibilă pe calendarul actorului.

**Diagrama de evenimente:**

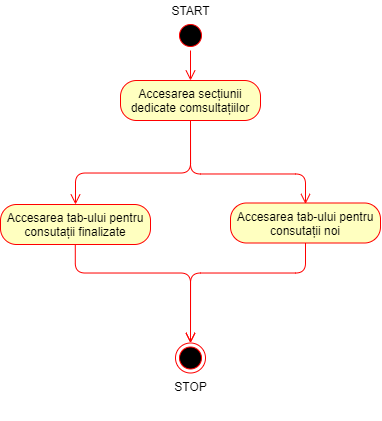


Figura 5.10 Diagrama de evenimente ”Vizualizarea consultațiilor atribuite”

##### Configurarea unei consultații

**Pașii scenariului:**

1. Actoul accesează secțiunea destinată consultațiilor
2. Actorul accesează tab-ul corespondent consultațiilor noi
   1. Dacă nu există consultații care pot fi configurate (deja finalizate), atunci scenariul se sfârșește odată cu acest pas
   2. Dacă există consultații noi, actorul poate continua cu pasul de configurare a uneia dintre acestea
3. Actorul accesează una dintre consultațiile noi
4. Actorul adaugă informații în cadrul consultației (observații, diagnostice, prescripții)
5. Actorul în pasul de finalizare al consultației
   1. Dacă actorul alege să nu finalizeze consultația, datele acesteia nu vor apărea ca fiind completate, iar consultația nu se va finaliza.
   2. Dacă actorul alege să finalizeze consultația, aceasta entitate este salvată și modificarea acesteia nu mai este posibilă.

**Precondiții:**

1. În primul rând, pentru a avea o programare în calendar (vizibilă de către angajații cabinetului), este nevoie să existe o o solicitare de programa, care poate fi obținută dacă fie pacientul solicită o programare, fie o astfel de entitate este creată de către un asistent medical.
2. Odată având o solicitare de programare, doctorul este nevoit să o accepte pentru ca entitatea reprezentând o consultație să fie creată. Din moment ce există o consultație creată, doctorul are acces la aceasta, fiind capabil să o configureze.

**Postcondiții:**

1. După configurare și salvare, consultația devine read-only, detaliile acesteia putând fi vizualizate (atât de angajații cabinetului, cât și de pacienți), dar fără posibilitatea de editare.
2. Dacă doctorul alege să atribuie și o prescripție în procesul de configurare al consultației, această acțiune va rezulta în trimiterea unui email pacientului conținând informații despre cel mai apropiat punct farmaceutic la care acesta poate achiziționa medicamentele prescrise.

**Diagrama de evenimente:**

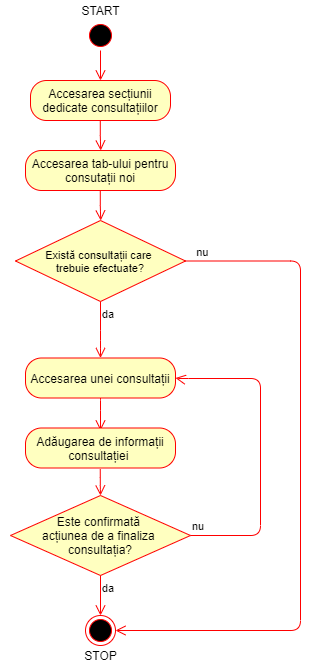


Figura 5.11 Diagrama de evenimente ”Configurarea unei consultații”

### Cazurile de utilizare ale rolului de pacient

Aplicația dezvoltată conține un modul care este destinat pacientului, conținând un număr de trei funcționalități specifice acestui rol.

##### Vizualizarea și editarea informațiilor personale

Pașii acestui scenariu sunt exact ca și cei pentru scenariul în care pacientul dorește să își editeze informațiile personale de natură medicală.

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată profilului personal
2. Actorul își vizualizează datele personale
   1. Dacă actorul alege să nu execute modificări, atunci scenariul ajunge la final
   2. Dacă actorul alege să își modifice datele, atunci scenariul continuă
3. Actorul este în pasul de confirmare a modificărilor făcute
   1. Dacă noile date salvate, modificările făcute se vor pierde
   2. Dacă actorul alege să își salveze modificările, datele acestuia vor fi actualizate cu noile valori introduse

**Postcondiții:**

1. Odată ce datele sunt salvate cu success, actorul va fi capabil să își vizualizeze datele, acestea având ultimele valori salvate de către acesta.
2. Editarea datelor personale de natură medicală va fi vizibilă și pentru angajații cabinetului medical la care este înregistrat actorul.

**Diagrama de evenimente:**

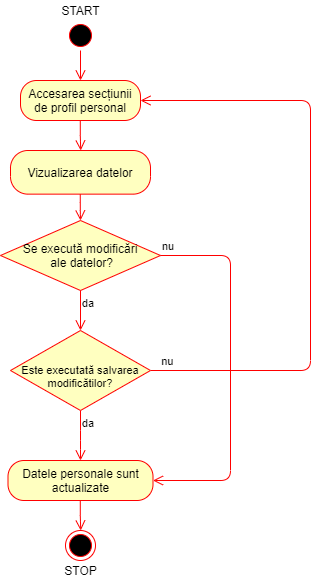


Figura 5.12 Diagrama de evenimente a cazului de utilizare ”Vizualizarea și editarea informațiilor personale”

##### Solicitarea de programări la cabinetul medical

**Pașii scenariului:**

1. Actorul accesează meniul destinat programărilor
2. Actorul optează pentru crearea unei noi programări
3. Actorul alege tipul de programare dorit
4. Actorul selectează o zi preferată pentru efectuarea programării
   1. Dacă este necesară alegerea unei alte zile, actorul va selecta o altă zi
   2. Dacă ziua selectată este cea dorită, actorul poate continua (pasul 5)
5. Actorul accesează meniul de confirmare programării
   1. Dacă actorul confirmă programarea, acesta este anunțat printr-un mesaj pe pagină că programarea a fost create
   2. Dacă actorul renunță să confirme programarea, acesta este redirecționat pe pagina destinată programărilor

**Postcondiții:**

1. Pacientul poate să vizualizeze programarea creată în lista de programări personale, care poate fi găsita pe pagina specifică din aplicație

Odată ce utilizatorul confirmă crearea programării, aceasta va fi afișată, împreună cu programările anterioare, pe pagina special destinată acestora.

1. Asistenții cabinetului vor vedea solicitarea

Gestionarea programărilor pentru cabinet reprezintă una dintre funcționalitățile la care are acces rolul de asistent medical. Oricând un asistent sau un pacient creează o cerere de programare, aceasta apare pe calendarul săptămânal al doctorului, conținând detalii precum: pacientul, doctorul catre care este directionata aceasta cerere, tipul si data programării.

1. Solicitarea de programare apare in calendarul saptamanal al doctorului selectat

Din moment ce programarile sunt elemente care leaga rolurile de pacient si de doctor,

calendarul unui doctor va afisa toate programarile, atat cele create de asistenti, cat si cele

create de pacienti.

**Puncte de extensie ale scenarului:**

1. Primirea de notificări prin email

În momentul în care doctorul cabinetului va lua o anumită acțiune în legătură cu programarea solicitată de către pacient (fie o acceptă, fie o refuză), pacientul va primi un email prin care va fi anunțat în legătură cu decizia care a fost luată de către medic.

Dacă o consultație este refuzată, ea nu va mai putea fi efectuată în intervalul selectat inițial, crearea unei noi solicitări pentru o consultație fiind soluția pentru a putea obține o programare.

Dacă o consultație este acceptată, actorul va primi un email în care i se vor reitera ziua și intervalul orar în care consultația va avea loc pentru ca acesta să sosească la cabinetul medical în momentul stabilit inițial.

1. Obținerea de indicații către cea mai apropiată farmacie

În cazul în care programarea a fost acceptată , aceasta devine, pentru rolul de doctor, o entitate de tipul ” consult”. În momentul în care pacientul se prezintă la ora stabilită prin programare la cabinet, doctorul începe consultul, iar, dacă în urma configurătii consultului, doctorul atașează o prescripție fișei de consult, pacientul va primi indicații prin email către cea mai apropiată farmacie care deține în stoc medicamentele prescrise. Acestă funcționalitate este simulată în cadrul aplicației realizate, datele farmaciilor find fictive.

**Diagrama de evenimente:**

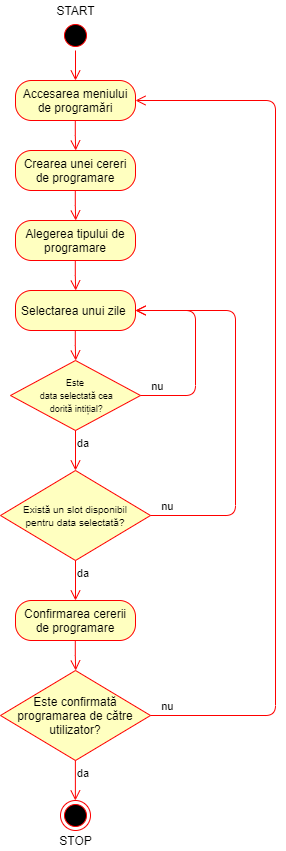


Figura 5.13 Diagrama de evenimente ”Solicitarea de programări la cabinetul medical”

##### Obținerea de indicații căre cel mai apropiat punct farmaceutic

**Pașii scenariului:**

1. Actorul primește un email care conține indicațiile pentru deplasarea înspre cel mai apropiat punct farmaceutic

Punctul farmaceutic este simulat în cadrul aplucației, dar se va considera că punctul farmaceutic ales există medicamentele prescrise de doctor în urma unui consult efectuat, pe care actorul le poate achiziționa.

Locația farmaciei va fi distribuită în cadrul emailului cu ajutorul Google Maps, utilizatorul fiind capabil să apese pe link-ul primit și ruta de la locația actuală până la punctul farmaceutic va fi configurată.

**Precondiții:**

1. Singura precondiție pentru ca actorul să poată fi parte a acestui scenariu necesită participarea acestuia la un consult, în cadrul căruia doctorul alege să atașeze o prescripție. Odată ce această acțiune are loc, mecanismul de trimitere de email se declanșează.

**Diagrama de evenimente:**

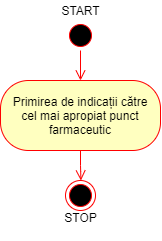


Figura 5.14 Diagrama de evenimente ”Obținerea de indicații căre cel mai apropiat punct farmaceutic”

##### Vizualizarea datelor legate de consultații

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată consultațiilor
   1. Dacă există consultații, actorul poate să acceseze una dintre consultațiile din lista afișată
   2. Dacă nu există consultații pentru actorul autentificat
2. Actorul vizualizează detaliile unei consultații

**Diagrama de evenimente:**

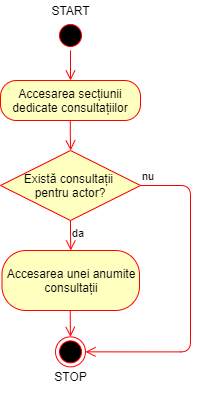


Figura 5.15 Diagrama de evenimente ”Vizualizarea datelor legate de consultații”

### Cazurile de utilizare ale rolurilor administrative

Cazurile de utilizare ale celor două tipuri de administratori sunt identice ca pași ai scenariului, singura diferență constând din faptul că administratorul de cabinet are un access mai restrâns, fiind capabil să gestioneze strict datele cabinetului la care a fost asignat și ale angajaților acestuia (asistenți medicali și doctori). Diferit de administratorul de cabinet, administratorul de aplicație poate să își execite un control absolut asupra tuturor cabinetelor și angajaților care sunt înregistrați în cadrul aplucației. În continuare, se vor aborda cazurile de utilizare comune, cu precizarea că un administrator de aplicație ar putea executa oricare dintre cele două operații pe oricare dintre cabinetele sau utilizatorii (strict asistenți medicali și doctori) înregistrați în aplicație.

#### Funcționalități comune ambelor tipuri de administratori

##### Adăugarea și editarea informațiilor legate de utilizatorii cabinetului (sau ai unui cabinet anume)

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată programărilor
   1. Dacă actorul alege să creeze un utilizator nou, atunci următorul pas al scenariul se va regăsi la numărul 3
   2. Dacă actorul alege să vizualizeze (și să posibil modifice) datele unui utilizator, atunci acesta va continua scenariul cu pasul numărul 2
2. Actorul alege să vizualizeze un utilizator din listă
   1. Dacă actorul nu dorește să modifice datele utilizatorului pe care a ales să le vizualizeze, atunci scenariul se sfârșește după această acțiune
   2. Dacă actorul dorește să modifice datele utilizatorului, scenariul continuă cu pasul 3
3. Actorul este în pasul de confirmare a modificărilor făcute
   1. Dacă actorul alege să nu salveze modificările făcute și părăsește pagina, datele introduse de acesta nu se vor salva, utilizatorul având aceleași date ca și la începerea acestui scenariu sau, în cazul creării unui utilizator nou, această entitate nu va putea fi salvată
   2. Dacă actorul alege să salveze modificările făcute, datele unui utilizator existent se vor actualiza, sau, în cazul creării unui utilizator nou, aceasta va fi salvat

**Postcondiții:**

1. Datele utilizatorului sunt actualizate (dacă se aplică scenariul de adăugare al unui nou utilizator)

Odată ce actorul modifică datele utilizatorului și alege să își salveze modificările, acestea vor fi vizibile atât pentru angajatul cabinetului (doctor sau asistent – asupra căruia s-au efectuat modificările de date), cât și pentru rolurile administrative.

**Puncte de extensie:**

1. Din moment ce un nou utilizator este creat, acesta își va putea accesa contul

Utilizatorul va primi un email cu datele de autentificare, aceastea constand din email-ul său și o parolă, care va fi generată la momentul creării contului.

1. Odată creat, utilizatorul va apărea în lista utilizatorilor, vizibilă atât pentrua administratorul cabinetului pentru care a fost creat utilizatorul, cât și pentru administratorul de aplicație

**Diagrama de evenimente:**

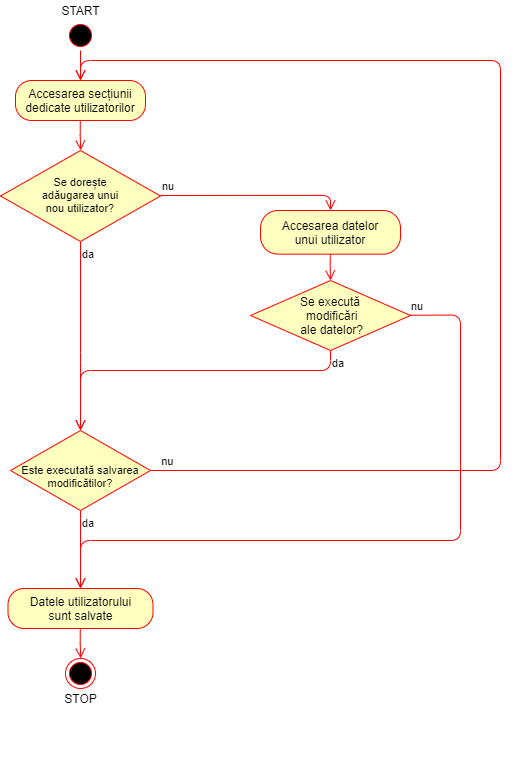


Figura 5.16 Diagrama de evenimente ”Adăugarea și editarea informațiilor legate de utilizatori”

##### Editarea informațiilor legate de cabinet(e)

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată cabinetelor
2. Actorul alege să vizualizeze datele unui cabinet specific (dacă actorul are rolul de administrator de aplucație, acesta va puea vizualiza datele cabinetului pe care îl administrează)
   1. Dacă actorul nu dorește să modifice datele cabinetului, scenariul se sfârșește după această acțiune
   2. Dacă actorul dorește să modifice datele cabinetului, scenariul continuă
3. Actorul este în pasul de confirmare a modificărilor făcute (dacă există)
   1. Dacă actorul alege să nu salveze modificările făcute și părăsește pagina, datele introduse de acesta nu se vor salva, cabinetul având aceeași configurație ca și la începerea acestui scenariu
   2. Dacă actorul alege să salveze modificările făcute pentru datele cabinetului, datele acestuia se vor actualiza

**Postcondiții:**

1. Odată ce actorul modifică datele cabinetului și alege să își salveze modificările, acestea vor deveni vizibile pentru ambele tipuri de roluri administrative.

**Diagrama de evenimente:**

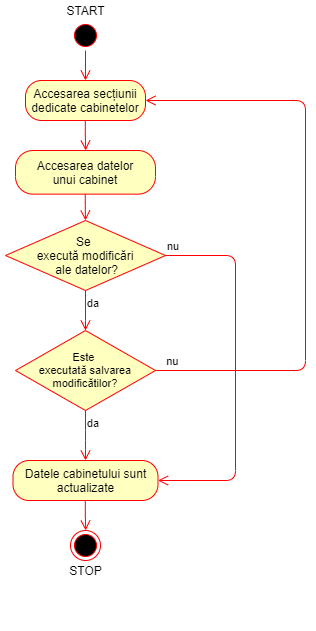


Figura 5.17 Diagrama de evenimente.”Editarea informațiilor legate de cabinet(e)”

#### Funcționalitatea specifică administratorior de aplicații

##### Definirea unui nou cabinet

**Pașii scenariului:**

1. Actorul accesează secțiunea destinată cabinetelor
2. Actorul optează pentru crearea unui cabinet nou
3. Actorul începe acțiunea de a introduce datele aferente cabinetului
   1. Dacă datele introduse nu sunt valide, câmpurile care conțin astfel de date vor fi semnalate corespunzător.
   2. Dacă datele sunt valide, scenariul poate continua cu următorul pas
4. Actorul este în pasul de confirmare a acțiunii de creare a unui nou cabinet
   1. Dacă date noului cabinet nu sunt salvate, modificările făcute nu vor putea fi înregistrate
   2. Dacă actorul alege să își salveze modificările, o nouă entitate de cabinet va fi creată în aplicație, datele acestuia cont vor fi cele introduse anterior de actor

**Postcondiții:**

1. După ce actorul introduce date valide, confirmă acțiunea de a crea un nou cabinet și este notificat de sistem că acțiunea a putut fi executată, un nou intrare de tipul ”cabinet” va fi vizibilă în cadrul secțiunii de cabinete a aplicației

**Puncte de extensie ale scenarului:**

1. Opțiunea de configurare a cabinetului devine disponibilă

Administratorul de aplicație poate crea (sau asgina un administrator de cabinet existent) pentru cabinetul creat anterior. Acesta din urmă este capabil să configureze cabinetul și, odată cu adăugarea de angajați, aceștia își pot adăuga pacienți și întregul spectru al aplicației.

**Diagrama de evenimente:**

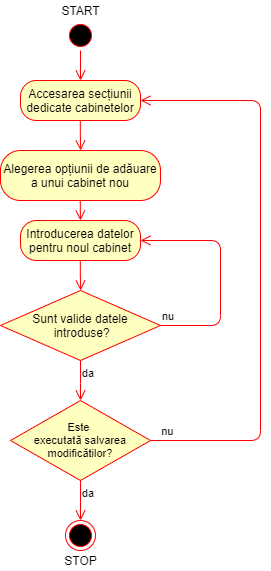


Figura 5.18 Diagrama de evenimente ”Definirea unui nou cabinet”

## Descrierea interacțiunii cu utilizatorul

### Diagrame de activitate

### Diagrame de secvență

### Diagrame de comunicare

## Arhitectura bazei de date

Diagrama bazei de date conține un număr de 16 tabele. În continuarea acestei secțiuni se poate consulta diagrama integrală, cât și o scurtă descrierere pentru fiecare dintre tablele componente.

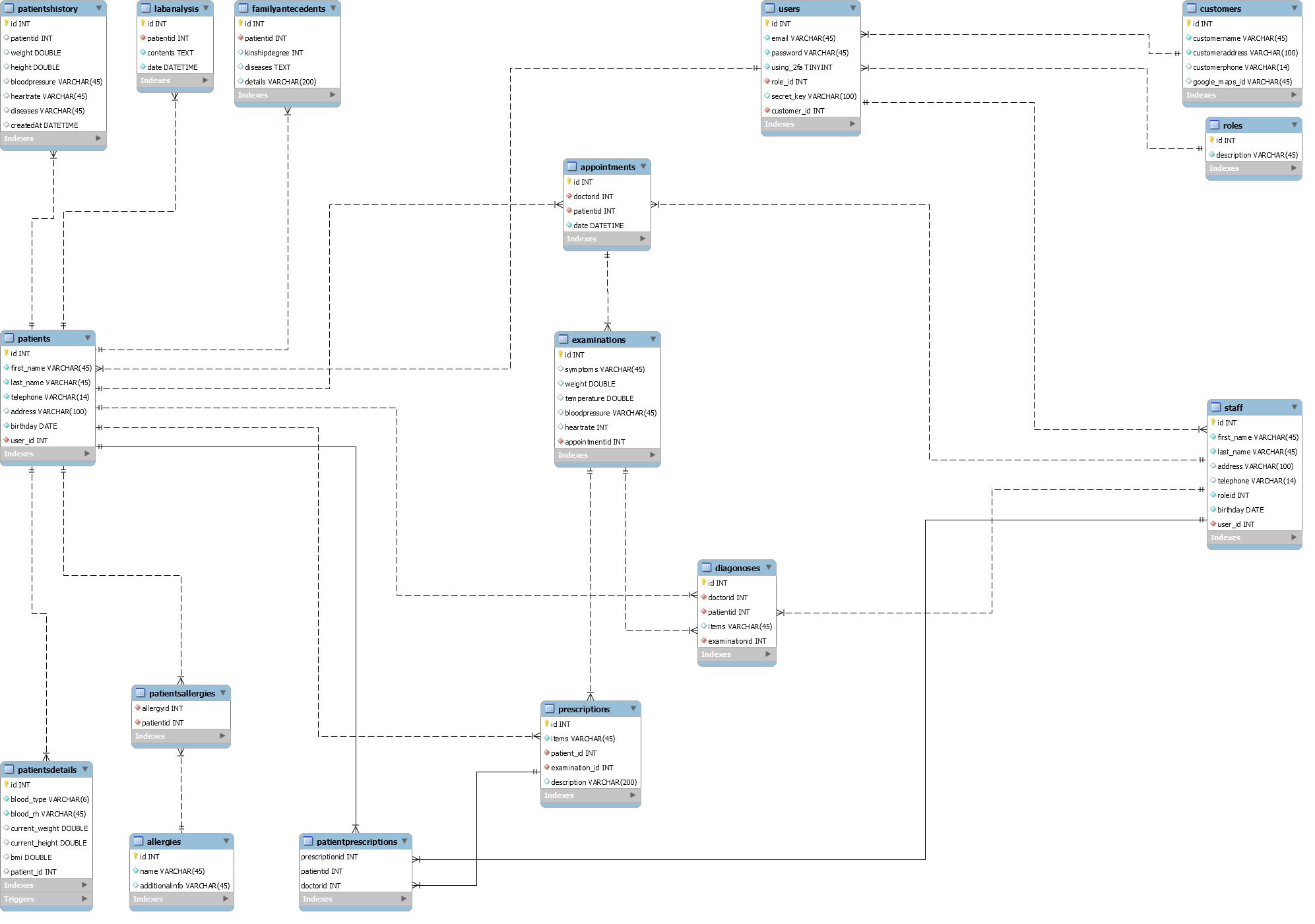


Figura 5.19 Diagrama bazei de date

TO DO: Descrierea fiecărui tabel: 1-2 rânduri

## Decizii de implementare

### Diagrame de clase

### Diagrame de pachete

### Diagrame de componente

### Diagrame de deployment

TO DO: De adăugat și descris după ce codul este finalizat și rafinat

# Testare şi Validare

În cadrul acestui capitol se vor prezenta scenariile executate în cadrul aplicației, accentul punându-se pe modul în care utilizatorii interacționează cu aplicația, cât și răspunsul sistemului la acțiunile executate de aceștia.

## Scenariile testate în cadrul aplicației

Scenariile prezentate vor fi ordonate astfel încât să poată simula modul în care un cabinet medical și pacienții săi colaborează.

În continuare, se va presupune că nu există pacienți ai cabinetului, acesta fiind creat foarte recent de către administratorul de aplicație și cel al cabinetului.

TO DO: Descriere flow ca o suită de scenarii la care sunt supusi toti actorii si toate functionalitatile aplicatiei

De la premisa ca nu exista un cabinet se va ajunge pana la existenta cabinetului, cu angajati si pacienti

Pașii

* 1. Admin aplicație: creează cabinetul + cont admin cabinet (pe mail, deci nu în aplicație)
  2. Admin cabinet: customizează cabinetul + adaugă doctor + adaugă asistenți medicali
  3. Doctor/asistent medical: actualizare informații personale + înregistrează pacienți
  4. Pacienții: actualizare date personale + adăugare date medicale + solicitare programări

APOI:

Ping-pong: Pacient -> Cabinet -> Pacient -> Cabinet

->>> Programari, calendar, edit-uri, operatii pe programari, primire de email-uri, generare de rapoarte pentru pacient (fisa, raport al consultatiilor, etc), incepere consulatie, adaugare diagnostic, adaugare prescriptie, primire de email catre cea mai apropiata farmacie, etc..

# Manual de Instalare si Utilizare

## Manual de instalare

### Cerințe hardware și software

În cadrul aceste secțiuni vor fi descrise cerințele pe care un sistem trebuie să le îndeplinească pentru a rula aplicația și pentru a o face disponibilă unui cabinet medical.

Datorită folosirii framework-ului Spring al Java nu este nevoie de un web server dedicat pentru a lansa aplicația, astfel că aceasta va putea fi lansată direct din IDE (engl. *Integrated Development Environment*). IDE-ul folosit (și recomandat pentru rularea aplicației) fiind Intellij IDEA, care poate fi descărcat de la adresa [aceasta](https://www.jetbrains.com/idea/download/#section=windows).

De asemenea, aplicația necesită un server pentru baza de date, iar cel folosit în cadrul acestei aplicații este MySQL Workbench de la Oracle și poate fi descărcat de la adresa [aceasta](https://dev.mysql.com/downloads/workbench/).

Luând în calcul resursele de care cele două aplicații au nevoie pentru a funcționa [28] [29], înainte de instalarea programelor referite anterior, viitorul utilizator trebuie să se asigure că stația de lucru suportă următoarea configurație:

* Procesor Intel Core sau Xeon sau un echivalent al tipului AMD CPU
* Număr de nuclee: Single (Recomandare: Dual sau Quad Core)
* Memorie RAM: 4 GB (Recomandare: 6 GB)
* Spațiu pe disc: aproximativ 5 GB
* Rezoluția ecranului de minim 1280x1024 (Recomandare: 1920x1200)
* Sistem de operare (versiunea minimă):
  + Windows: Windows 8
  + MacOS: 10.13
  + Linux: Orice distribuție care suportă Gnome, KDE sau Unity DE (DE - engl. *Desktop Environment*)

Totodată, utilizatorul are nevoie de anumite programe auxiliare, precum Git și Visual Studio Code. Primul dintre acestea poate fi descărcat de [aici](https://gitforwindows.org/) și este utilizat pentru preluarea repository-ul în care se regăsește aplicația. Cel de-al doilea program este un editor de text opțional (dar recomandat) care poate fi folosit pentru lansarea serverului de front-end al aplicației, programulse poate descărca de la adresa [acesta](https://code.visualstudio.com/download). Dacă nu se dorește instalarea unui software suplimentar, lansarea modulului de front-end se poate inițializa și prin intermediului IntelliJ IDEA.

### Inițializarea aplicației

#### Obținerea aplicației

După instalarea tuturor programelor necesare, utilizatorul poate continua cu pasul în care obține propriu-zis accesul la aplicația dezvoltată. Acest proces poate fi realizat prin urmarea pașilor de mai jos:

1. Se navighează în sistemul de fișiere al calculatorului și se alege locația în care se va afla codul aplicației
2. Selectând viitorul director al aplicației, se acționează click-dreapta și se alege opțiunea *Git Bash here*
3. În terminal, se execută comanda *git clone*, argumentul acesteia fiind [link](https://github.com/Nebryth/proiect-licenta-medapp.git)-ul către repository-ul aplicației
4. Închiderea ferestrei *Git Bash* după ce clonarea a fost încheiată cu success

După executarea tuturor pașilor, utilizatorul ar trebui să aibă acces la toate fișierele proiectului (documentația cu resursele acesteia, cât și codul complet al aplicației), care vor avea o structură asemănătoare cu cea din următoarea imagine.

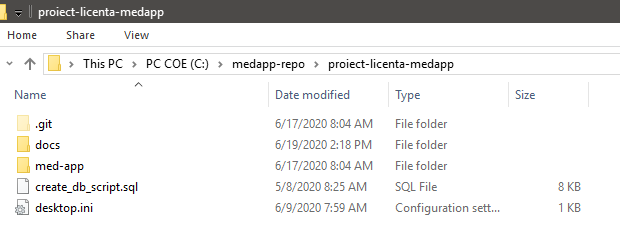


Figure 7.1 Structura de fișiere a proiectului

#### Lansarea aplicației

Pentru a accesa aplicația, aceasta trebuie să fie lansată, acțiune ce se realizează urmând pașii următori:

TODO: Adăugare de pași și screenshot-uri

## Manual de utilizare

TODO: Adăugare de printscreen-uri cu săgeți și explicații:

# Concluzii

## Analiza rezultatelor obținute

Scopul proiectului realizat a constat în realizarea unei aplicații care se adresează cabinetelor medicale, oferind funcționalitate atât pentru membrii cabinetului (asistenți medicali, doctori și administrator de cabinet), cât și pentru pacienții înregistrați la cabinetul respectiv. Pentru diferitele roluri, aplicația prezintă atât funcționalități comune, cât și unele specifice; membrii cabinetului au access la datele generale ale mai multor pacienți, putând seta și modifica statusurile programărilor, de a iniția și configura consulturi, atașându-le diagnostice și prescripții. Totodată, pacienții au posibilitatea de solicita programări, de a-și vizualiza și edita datele de natură medicală și personală, de a crea și de a vizualiza datele consulturilor la care aceștia au participat, de a primi informații în legătură cu cel mai apropiat punct pentru preluarea medicamentelor prescrise de către cabinet. Aplicația este disponibilă atât de pe web, cât și de pe tabletă sau de pe smartphone, design-ul ei fiind unul versatil, adaptabil oricărui tip de dispozitiv folosit.

Aplicația poate fi accesată cu un minim de efort, folosind un browser desktop , cât și de pe alte dispozitive, design-ul acesteia fiind adaptabil oricarui tip de rezolutie si browser. Utilizatorii acestei aplicatii vor fi atat angajatii cabinetului, cat si pacientii inregistrati. Modulele aplicației prezintă atât funcționalități comune, cât și unele specifice; membrii cabinetului au access la datele generale ale mai multor pacienți, putând seta programări, crea consulturi, atașa diagnostice, prescrie rețete, iar pacienții au posibilitatea de solicita programări, de a-și vizualiza datele medicale personale, de a crea și analiza metrici în legătură cu anumite condiții personale, de a primi informații în legătură cu cel mai apropiat punct pentru preluarea medicamentelor prescrise de către cabinet, etc.

## Analiză comparativă

În cadrul acestei secțiuni se va realiza o analiză comparativă în care vor fi implicate aplicațiile existente pentru cabinetele medical (descrise în cadrul capitolului 3 al acestei lucrări) și aplicația dezvoltată.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | IcMED | MedSoft | MedXLine | Evolve | Med-App |
| Funcționalități | Funcționalitate 1 |  |  |  |  |  |
| Funcționalitate 2 |  |  |  |  |  |
| Funcționalitate 3 |  |  |  |  |  |
| Funcționalitate 4 |  |  |  |  |  |
| Funcționalitate 5 |  |  |  |  |  |
| Funcționalitate 6 |  |  |  |  |  |

TO DO: Completare tabel și concluzii

## Dezvoltări și îmbunătățiri ulterioare

TO DO: Dezvoltarea punctelor de mai jos

* Integrare cu CAS
* Integrare su software de farmacii
* Îmbunătățirea aplicației pentru a îngloba funcționalități de tipul PWA (engl. *Progressive Web Apps*)

Bibliografie

|  |  |
| --- | --- |
| [1] | Age UK, „The Internet and Older People in the UK – Key Statistics,” 1 July 2016. [Interactiv]. Available: https://www.ageuk.org.uk/globalassets/age-uk/documents/reports-and-publications/reports-and-briefings/active-communities/rb\_july16\_older\_people\_and\_internet\_use\_stats.pdf. [Accesat 1 June 2020]. |
| [2] | T. Soikkeli, J. Karikoski și H. Hämmäinen, „Diversity and end user context in smartphone usage sessions,” în *Department of Communications and Networking, Aalto University*, Espoo, 2011. |
| [3] | W. Xu și Y. Liu, „mHealth Apps: A Repository and Database of Mobile Health Apps,” *Journal of Medical Internet Research - mHealth and uHealth,* vol. 3, nr. 1, 2015. |
| [4] | research2guidance, „mHealth App Developer Economics,” The State of the Art of mHealth App Publishing, 2014. |
| [5] | M. Attaran, S. Attaran și D. Kirkland, „The Need for Digital Workplace: Increasing Workforce Productivity in the Information Age,” *International Journal of Enterprise Information Systems,* vol. 15, nr. 1, January-March 2019. |
| [6] | M. Yasini, J. Beranger, P. Desmarais, L. Perez și G. Marchand, „mHealth Quality: A Process to Seal the Qualified Mobile Health Apps,” IOS Press, 2016. |
| [7] | R. U. Pande, Y. Patel, C. J. Powers, G. D'ancona și H. L. Karamanoukian, „The Telecommunication Revolution in the Medical Field: Present Applications and Future Perspective,” *CURRENT SURGERY,* vol. 60, nr. 6, pp. 636-640, 2003. |
| [8] | M. N. K. Boulos, S. Wheeler, C. Tavares și R. Jones, „How smartphones are changing the face of mobile and participatory healthcare: an overview, with an example from eCAALYX,” BioMedical Engineering OnLine, 2011. |
| [9] | Y. Yang, M. K. Sullivan, P. P. Wang și K. D. Naidu, „Applications of Computer Simulation in Medical Scheduling,” Durham, NC, 2000. |
| [10] | P.-G. Svensson, „eHealth Applications in Health Care Management,” *eHealth International,* vol. 1, nr. 1, 2002. |
| [11] | W. Jobe, „Native Apps vs. Mobile Web Apps,” *International Journal of Interactive Mobile Technologies (iJIM),* vol. 7, nr. 4, pp. 27-32, 2013. |
| [12] | K. J. Kadhim, M. F. Odhaib, Y. H. Hadi, A. A. Mudhi, A. K. Abdlasahib și R. B. W. Sheik Osman, „Developing a Multi Platforms Web Application for Mobile Using HTML5,” *Journal of Information Technology & Software Engineering,* vol. 8, nr. 2, 2018. |
| [13] | K. Selvarajah, M. P. Craven, A. Massey, J. Crowe, K. Vedhara și N. Raine-Fenning, „Native Apps versus Web Apps: Which is Best for Healthcare Applications?,” HCI: International Conference on Human-Computer Interaction (Part II), Las Vegas, 2013. |
| [14] | S. R. Stoyanov, L. Hides, D. J. Kavanagh, O. Zelenko, D. Tjondronegoro și M. Mani, „Mobile App Rating Scale: A New Tool for Assessing the Quality of Health Mobile Apps,” *Journal of Medical Internet Research,* vol. 3, nr. 1, 2015. |
| [15] | ICMED, „ICMED | >> Home,” [Interactiv]. Available: http://icmed.ro/en/. [Accesat 18 June 2020]. |
| [16] | Google Play, „icMED.Mobile - Apps on Google Play,” [Interactiv]. Available: https://play.google.com/store/apps/details?id=eu.syonic.icmed.patient&hl=en. [Accesat 7 June 2020]. |
| [17] | MedSoft, „Suport - MedSoft,” [Interactiv]. Available: https://www.med-soft.ro/produse/. [Accesat 18 June 2020]. |
| [18] | Evolvo, „Evolvo România | Software medical online și aplicație medicală,” [Interactiv]. Available: https://evolvo.ro/. [Accesat 18 June 2020]. |
| [19] | Google Play, „Evolvo medical - Apps on Google Play,” [Interactiv]. Available: https://play.google.com/store/apps/details?id=ro.evolvo.app&hl=en. [Accesat 9 June 2020]. |
| [20] | App Store, „Evolvo medical on the App Store,” [Interactiv]. Available: https://apps.apple.com/ro/app/evolvo-medical/id1091587519. [Accesat 9 June 2020]. |
| [21] | MedXLine, „MedXLine - aplicatie de gestiune clinici, policlinici, cabinete medicale,” [Interactiv]. Available: https://www.medxline.ro/index.php. [Accesat 18 June 2020]. |
| [22] | Packt, „Benefits of the Spring Framework,” Packt, [Interactiv]. Available: https://subscription.packtpub.com/book/application\_development/9781783987368/1/ch09lvl1sec11/benefits-of-the-spring-framework#:~:text=Spring%20provides%20a%20lightweight%20container,productivity%20and%20reduces%20the%20error.. [Accesat 17 June 2020]. |
| [23] | Fusion SoftTech, „What’s New in Spring Boot 2.0?,” Fusion SoftTech - Idea To Action, 2 February 2020. [Interactiv]. Available: https://www.fusionsofttech.co.in/whats-new-in-spring-boot-2-0/. [Accesat 18 June 2020]. |
| [24] | Pluralsight, „HTML5 | Pluralsight,” Pluralsight, [Interactiv]. Available: https://www.pluralsight.com/paths/html5. [Accesat 18 June 2020]. |
| [25] | C. Skellu, „Upgrade Bootstrap 4 Alpha 6 to Beta - WDstack - Medium,” Medium - WDstack, 11 August 2017. [Interactiv]. Available: https://medium.com/wdstack/upgrade-bootstrap-4-alpha-6-to-beta-ca582f15ee32. [Accesat 18 June 2020]. |
| [26] | S. Varma, „Getting Started With React.js — Part 2 -Better Programming - Medium,” Better Programming - Medium, 24 October 2019. [Interactiv]. Available: https://medium.com/better-programming/getting-started-with-react-js-part-2-3dbd08c680a2. [Accesat 18 June 2020]. |
| [27] | B. Bozhanov, „Enabling Two-Factor Authentication for your Web Application,” DZone, 6 December 2017. [Interactiv]. Available: https://dzone.com/articles/enabling-two-factor-authentication-for-your-web-ap. [Accesat 18 June 2020]. |
| [28] | MySQL Workbench, „FAQ - Hardware/OS - The MySQL Workbench Developer Central Site,” [Interactiv]. Available: https://mysqlworkbench.org/faq/faq-3/. [Accesat 18 June 2020]. |
| [29] | JetBrains, „Install IntelliJ IDEA﻿ - Help | IntelliJ IDEA,” JetBrains, 15 June 2020. [Interactiv]. Available: https://www.jetbrains.com/help/idea/installation-guide.html. [Accesat 18 June 2020]. |
| [30] | R. Abela, „Setting up 2FA on WordPress with the Google Authenticator app,” Security Boulevard, 13 February 2020. [Interactiv]. Available: https://securityboulevard.com/2020/02/setting-up-2fa-on-wordpress-with-the-google-authenticator-app/. [Accesat 18 June 2020]. |
| [31] | icMED, „ICMED | Comunitate pentru sănătate,” icMED, [Interactiv]. Available: http://icmed.ro/. [Accesat 5 June 2020]. |

# Tabelul figurilor utilizate

[Figura 1.1 Utilizatori săptămânali ai internetului în 2017 (CEE) 1](#_Toc43794286)

[Figura 1.2 Utilizatori săptămânali ai internetului în 2018 (CEE) 1](#_Toc43794287)

[Figura 1.3 Utilizatori săptămânali ai internetului în 2019 (CEE) 2](#_Toc43794288)

[Figura 1.4 Tipurile de aplicații medicale din magazinele online [4] 3](#_Toc43794289)

[Figura 3.1 Simbolul *icMed* (Sursă: [15]) 11](#_Toc43794290)

[Figura 3.2 Simbolul *icMED.Mobile* (Sursă: [16]) 11](#_Toc43794291)

[Figura 3.3 Simbolul *MedSoft* (Sursă: [17]) 12](#_Toc43794292)

[Figura 3.4 Simbolul *Evolvo* (Sursă: [18]) 12](#_Toc43794293)

[Figura 3.5 Simbolul *MedXLine* (Sursă: [21]) 13](#_Toc43794294)

[Figura 4.1 Diagrama cazurilor de utilizare pentru rolul de asistent medical 17](#_Toc43794295)

[Figura 4.2 Diagrama cazurilor de utilizare pentru rolul de administrator de cabinet 18](#_Toc43794296)

[Figura 4.3 Diagrama cazurilor de utilizare pentru rolul de administrator de aplicație 18](#_Toc43794297)

[Figura 4.4 Simbolul Java Spring (Sursă: [23]) 18](#_Toc43794298)

[Figura 4.5 Simbolurile HTML5, Bootstrap și React (Surse: [24], [25], [26]) 19](#_Toc43794299)

[Figura 4.6 Simbolul Google Authenticator(Sursă: [9]) 19](#_Toc43794300)

[Figura 5.1 Diagrama de evenimente a cazului de utilizare ”Activarea și autentificarea folosind 2FA” 24](#_Toc43794301)

[Figura 5.2 Diagrama de evenimente a cazului de utilizare ”Vizualizarea și editarea informațiilor personale” 25](#_Toc43794302)

[Figura 5.3 Diagrama de evenimente ”Adăugarea și editarea informațiilor legate de pacienți” 26](#_Toc43794303)

[Figura 5.4 Diagrama de evenimnete ”Înregistrarea unui nou pacient în cabinet” 27](#_Toc43794304)

[Figura 5.5 Diagrama de evenimente ”Vizualizarea calendarului de programări” 28](#_Toc43794305)

[Figura 5.6 Diagrama de evenimente ”Generarea de rapoarte privind istoricul de consultații ale unui pacient” 29](#_Toc43794306)

[Figura 5.7 Diagrama de evenimente ”Adăugarea și editarea de programări” 30](#_Toc43794307)

[Figura 5.8 Diagrama de evenimente ”Generarea unei fișe a pacientului” 31](#_Toc43794308)

[Figura 5.9 Diagrama de evenimente ”Acceptarea sau refuzarea de programări” 32](#_Toc43794309)

[Figura 5.10 Diagrama de evenimente ”Vizualizarea consultațiilor atribuite” 33](#_Toc43794310)

[Figura 5.11 Diagrama de evenimente ”Configurarea unei consultații” 34](#_Toc43794311)

[Figura 5.12 Diagrama de evenimente a cazului de utilizare ”Vizualizarea și editarea informațiilor personale” 35](#_Toc43794312)

[Figura 5.13 Diagrama de evenimente ”Solicitarea de programări la cabinetul medical” 37](#_Toc43794313)

[Figura 5.14 Diagrama de evenimente ”Obținerea de indicații căre cel mai apropiat punct farmaceutic” 38](#_Toc43794314)

[Figura 5.15 Diagrama de evenimente ”Vizualizarea datelor legate de consultații” 38](#_Toc43794315)

[Figura 5.16 Diagrama de evenimente ”Adăugarea și editarea informațiilor legate de utilizatori” 40](#_Toc43794316)

[Figura 5.17 Diagrama de evenimente.”Editarea informațiilor legate de cabinet(e)” 41](#_Toc43794317)

[Figura 5.18 Diagrama de evenimente ”Definirea unui nou cabinet” 42](#_Toc43794318)

[Figura 5.19 Diagrama bazei de date 43](#_Toc43794319)

1. <https://w3.unece.org/PXWeb/en> [↑](#footnote-ref-1)
2. <https://www.ageuk.org.uk/> [↑](#footnote-ref-2)