

Document de cerințe

Platformă de Telemedicină

Analiza și Modelarea Sistemelor Software

Introducere

Context

În România, sistemul medical este adesea marcat de limitări care îngreunează accesul rapid la consultații și servicii de specialitate, în special în sectorul public, unde timpul de așteptare poate fi considerabil. Pentru mulți pacienți, accesul la servicii private, deși mai prompt, rămâne inaccesibil din cauza costurilor ridicate. Acest context al unui sistem public suprasolicitat și al costurilor ridicate din privat este punctul de plecare al ideii acestui proiect. Platforma propusă își propune să ofere o metodă rapidă și accesibilă prin care pacienții să poată înțelege mai bine gravitatea simptomelor lor și să primească o recomandare preliminară de specialitate, facilitând astfel un acces mai rapid la îngrijire medicală.

Domeniu de aplicare

Platforma propusă oferă o metodă alternativă menită să eficientizeze accesul la servicii medicale, cu funcționalități integrate pentru a îmbunătăți rapiditatea și accesibilitatea consultațiilor. Sistemul este prevăzut cu următoarele funcționalități esențiale:

- *Consultații online:* Permite desfășurarea consultațiilor medicale de la distanță, prin videoconferință sau chat.
- *Sugestie de specialitate medicală:* Sistemul permite sugestia unui domeniu, respectiv a unui set de specialiști asociați diagnosticului prezis pe baza simptomelor.
- *Extragerea simptomelor din PDF:* Automatizează extragerea simptomelor din documente medicale încărcate (PDF), accelerând astfel procesul de triere.
- *Stocarea programărilor și a istoricului:* Permite gestionarea programărilor și arhivarea istoricului medical într-un mod organizat și securizat.

Cerințe funcționale

Consultări online

- *Funcționalitate principală:* Sistemul va permite utilizatorilor să programeze și să desfășoare consultații online printr-un call video.
- *Detalii:*
 - Pacientul poate crea o programare direct din aplicație sau pe baza analizei simptomelor introduse de utilizator. Dacă pacientul alege să își introducă simptomele în platformă, aceasta va sugera o listă de medici specialiști relevanți, în funcție de domeniul medical corespunzător afecțiunii probabile. Pacientul va avea opțiunea de a alege și programa o consultație cu unul dintre specialiști, în funcție de preferințele sale.
 - Platforma va include un sistem de notificări pentru a le reaminti utilizatorilor de consultațiile viitoare.

- Consultațiile se vor desfășura printr-un sistem de videoconferință integrat în platformă.
 - *Detalii tehnice:*
 - Platforma de videoconferință a fost realizată în Node.js, utilizând WebRTC ca suport pentru comunicarea în timp real și librăria Socket.IO pentru mecanismul de signaling.
1. Configurarea serverului HTTPS a fost realizată utilizând Express.js și un certificat auto-semnat. Un mediu securizat este necesar pentru stabilirea unei conexiuni de tip WebRTC, deoarece accesul la dispozitivele media și conexiunea peer-to-peer sunt permise doar printr-o conexiune securizată.
 2. Integrarea WebRTC
WebRTC asigură conexiunea peer-to-peer și permite transferul de fluxuri media între utilizatori, fără a necesita un server separat. Procesul de integrare presupune:
 - **Obținerea fluxurilor media:** aplicația solicită accesul la camera și microfonul utilizatorului prin API-ul ***getUserMedia***
 - **Crearea conexiunii peer-to-peer:** conexiunea directă între clienți se stabilește utilizând obiectul ***RTCPeerConnection***
 - **Partajarea fluxurilor media:** fluxurile locale sunt adăugate la conexiune prin metoda ***addTrack*** și partajate către celălalt client
 - **Gestionarea evenimentelor callback:** fluxurile media remote, candidații ICE și starea conexiunii sunt gestionate prin evenimente de tip callback, declanșate atunci când clientul local primește un răspuns de la clientul remote
 3. Signaling Server
Mecanismul de signaling este gestionat la nivelul serverului web, extins cu Socket.IO. Rolul acestuia este de a facilita schimbul de mesaje între utilizatori pentru inițierea și configurarea conexiunii peer-to-peer.
Fluxul semnalizării include schimbul ofertelor și răspunsurilor SDP pentru negocierea conexiunii și transmiterea candidaților ICE între cei doi clienți pentru stabilirea rutei de conectare.
 4. STUN Server
Un astfel de server este utilizat pentru a descoperi adresele IP publice ale utilizatorilor aflați în spatele unui NAT/firewall. Acesta facilitează, cu ajutorul candidaților ICE, găsirea unei rute pe care cei doi clienți se pot conecta. Candidații ICE reprezintă diverse metode de adresare (IP-uri și porturi) pe care un client le poate folosi pentru a se conecta cu un alt

client. Aplicația utilizează două dintre serverele STUN oferite de Google, ***stun:stun.l.google.com:19302*** și ***stun:stun1.l.google.com:19302***

5. Fluxul întregii aplicații

- **Conectarea la serverul web:** Un client inițiază o conexiune către server și se conectează printr-un socket pentru semnalizare
- **Obținerea fluxurilor media locale:** Clientul permite accesul la cameră și microfon, fluxurile media sunt capturate
- **Inițierea conexiunii peer-to-peer:** Clientul creează o ofertă pe care o trimite celui alt utilizator prin serverul de signaling, apelează la serverul STUN pentru a genera candidați ICE și adaugă fluxurile locale la conexiune
- **Negocierea conexiunii:** Clientul care primește oferta trimite o ofertă de tip răspuns către clientul care a inițiat apelul, se realizează schimbul de candidați ICE pentru stabilirea conexiunii directe
- **Transmiterea fluxurilor media:** Fluxurile media sunt transmise direct între cei doi clienți prin conexiunea peer-to-peer, evenimentele callback de tip ***track*** setează în browser-ul clientului local fluxurile media primite de la clientul remote
- **Deconectarea:** Când un utilizator închide apelul video, redarea fluxurilor media este oprită și conexiunea închisă

Sugestie medic bazată pe AI

- **Funcționalitate principală:** Sistemul utilizează un algoritm de inteligență artificială care analizează simptomele introduse de utilizator și generează o predicție preliminară asupra posibilei afecțiuni. Pe baza acestei predicții, platforma sugerează o listă de specialiști în domeniul medical relevant pentru a facilita alegerea unui medic potrivit.
- **Detalii:**
 - Utilizatorii își descriu simptomele detaliate, separate prin virgulă, într-un textbox, în limba engleză. (de exemplu: Enlarged, swollen veins, often in the legs)
 - Algoritmul AI analizează simptomele introduse și identifică un domeniu de specialitate medicală potrivit (de exemplu: cardiologie, dermatologie, neurologie).
 - Sistemul oferă utilizatorului o listă de medici specialiști relevanți în funcție de predicția afecțiunii.
 - După revizuirea sugestiilor, utilizatorul poate alege și programa o consultație online cu specialistul dorit.
- **Detalii tehnice:**
 1. **Modelul AI:**

- *Model utilizat:* T5-small – un model de tip Seq2Seq care este folosit pentru generarea de texte pe baza unor inputuri textuale.
- *Set de date:* QuyenAnhDE/Diseases_Symptoms – un set de date care conține simptome și afecțiuni asociate, utilizat pentru antrenarea modelului.
- *Fluxul de antrenare al modelului AI:*
 - Datele sunt preprocesate și tokenizate folosind AutoTokenizer.
 - Modelul AI este antrenat folosind Seq2SeqTrainer, pe 1000 de epoci.
 - După antrenament, modelul este salvat pentru a fi utilizat în producție.
- 2. *Sugestii de medici:*
 - *Folosirea unui fișier CSV:* Sistemul utilizează un fișier CSV (Doctor_Versus_Disease.csv) care conține o mapare între denumirile bolilor și specializările medicale asociate.
 - *Interogarea bazei de date:* După ce afecțiunea este prezisă de model, sistemul extrage specializarea relevantă și caută medici în baza de date. Utilizând interogări SQL, sunt selectați medicii care se încadrează în specializarea respectivă, iar lista acestora este returnată utilizatorului.

Extragerea simptomelor din format PDF

- *Funcționalitate principală:* Platforma va permite utilizatorilor să încarce fișiere PDF cu rapoarte medicale, iar sistemul va extrage automat simptomele relevante.
- *Detalii tehnice:*
 - *Procesul de extragere a simptomelor din fișiere PDF:*
 - *Biblioteca utilizată:* Algoritmul utilizează PyPDF2 pentru a citi și extrage textul din fișiere PDF. Textul extras este apoi procesat pentru a identifica secțiunile relevante cu simptomele.
 - *Reguli de căutare a simptomelor:* Sistemul folosește expresii regulate (re.search) pentru a căuta o secțiune specifică din document care conține cuvântul "SYMPTOMS". După identificarea acestei secțiuni, algoritmul caută și extrage simptomele enumerate, ignorând orice text care nu este relevant.
 - *Curățarea și formatarea simptomelor:* Simptomele sunt extrase și procesate pentru a elimina eventualele caractere necorespunzătoare sau semne de punctuație (de exemplu, liniuțe sau spații suplimentare).
 - *Formatul așteptat al documentului:* Algoritmul este conceput să proceseze fișiere PDF care respectă un template specific de raport medical, în care secțiunea de simptome începe cu cuvântul „SYMPTOMS” și continuă cu o listă de simptome separate prin linii noi.
- *Detalii:*
 - Utilizatorii vor încărca documente PDF într-un formular.
 - Algoritmul de procesare a textului va analiza documentul și va identifica simptomele relevante.
 - Simptomele extrase vor fi prezentate utilizatorului pentru verificare.

Stocarea și gestionarea datelor medicale

- *Funcționalitate principală:* Platforma va permite stocarea sigură a datelor despre consultații și a istoricului medical al pacienților.
- *Detalii:*
 - Datele vor fi criptate și stocate într-un mediu sigur, conform standardelor de securitate.
 - Utilizatorii vor avea posibilitatea de a accesa și gestiona istoricul consultațiilor și al diagnosticului.
 - Accesul la dosarele medicale va fi permis doar medicilor autorizați și pacienților respectivi.

Cerințe non-funcționale

Securitate

- Datele vor fi criptate în tranzit și în repaus.
- Sistemul va include autentificare pentru accesul utilizatorilor și al medicilor.

Utilizabilitate

- Platforma trebuie să fie intuitivă și ușor de utilizat atât pentru medici, cât și pentru pacienți, indiferent de nivelul de cunoștințe tehnice sau medicale al fiecăruia.
- Interfața va fi accesibilă pe dispozitive web.

Condiții și restricții

- Platforma trebuie să fie compatibilă cu principalele browsere web.
- Sistemul va permite accesul doar medicilor certificați, ale căror permisiuni vor fi verificate periodic.

Specificații ale interfeței utilizatorului

Pagina principală (Dashboard)

- *Pentru pacienți:*
 - Secțiune de acces pentru programarea unei noi consultații online.
 - Acces la istoricul consultațiilor și la dosarul medical.
- *Pentru medici:*
 - Acces la lista de consultații programate.
 - Secțiune de vizualizare a dosarului medical al pacienților consultați anterior.
 - Sistem de notificări pentru consultațiile viitoare.
 - Acces rapid la istoricul interacțiunilor cu pacienții și opțiunea de a scrie notițe sau diagnostice.

Pagina de consultație online

- Fereastră de videoconferință integrată.

Pagina de sugestie de specialitate medicală

- Introducerea simptomelor manual (ex. „Fever, Cough”) sau încărcarea unui fișier PDF pentru extragerea simptomelor.
- Procesarea simptomelor și generarea unui diagnostic preliminar.
- Sugestii de specializație și medici relevanți pe baza diagnosticului generat.
- Posibilitatea de a programa o consultație direct cu medicii sugerați.
- Alerte care indică faptul că diagnosticul este orientativ și nu înlocuiește consultul medical real.

Setări și cont utilizator

- Opțiuni de gestionare a informațiilor contului (e.g., date personale, detalii de contact).

Designul Sistemului

În cadrul acestui proiect, sunt utilizate două patternuri de design principale: *Strategy* și *Singleton*. Acestea sunt implementate pentru a asigura modularitatea și flexibilitatea sistemului, respectiv pentru a sprijini gestionarea logicii aplicației și interacțiunile utilizatorilor.

Modelul Singleton pentru conexiunea la baza de date

Implementarea conexiunii la baza de date folosind un design pattern de tip *Singleton* este o soluție eficientă și elegantă pentru gestionarea resurselor. Acest model asigură că există o singură instanță a conexiunii, partajată în întreaga aplicație, îmbunătățind astfel eficiența prin eliminarea conexiunilor redundante, asigurând consistența operațiunilor asupra bazei de date și centralizând configurarea.

Detalii de implementare:

Singleton-ul este realizat prin suprascrierea constructorului, care controlează crearea instanței. La fiecare cerere de instanțiere a clasei, se verifică dacă aceasta a fost deja creată. Dacă există o instanță existentă, aceasta este reutilizată. Altfel, o nouă instanță este generată și salvată în variabila partajată `_instance`, iar conexiunea propriu-zisă este inițializată la primul apel al metodei `get_connection`.

Modelul Strategy pentru logica AI

Platforma utilizează patternul *Strategy* pentru gestionarea componentelor de Inteligență Artificială (AI), oferind flexibilitate în înlocuirea și adaptarea rapidă a algoritmilor de procesare a simptomelor și generare a diagnosticelor. Aceasta permite integrarea ușoară a noilor modele AI, fără a afecta restul aplicației, asigurând astfel extensibilitate și scalabilitate. Designul separă logica de procesare a simptomelor de restul aplicației, facilitând întreținerea și testarea componentelor individuale. Astfel, platforma poate actualiza și îmbunătăți algoritmi de

diagnosticare pe măsură ce apar noi tehnologii sau cercetări, fără a întrerupe experiența utilizatorului.

Implementarea modelului Strategy include următoarele componente:

- Interfața *PredictionStrategy* definește o metodă comună pentru toate soluțiile de procesare AI, asigurând consistența interfeței de utilizare.
- Strategiile concrete (en. concrete strategies) reprezintă implementările concrete ale strategiilor de procesare a simptomelor, care implementează interfața *PredictionStrategy*. În acest caz, implementăm o strategie pentru *DiagnosisClassifier*, care folosește modelul T5.
- Clasa *AIContext* acționează ca un "context" care încarcă și utilizează o strategie concretă pentru a procesa simptomele și a genera recomandări.

