

# **SISTEM PENTRU MONITORIZAREA COMPLIANȚEI LA TRATAMENTUL MEDICAMENTOS AMBULATORIU**

**Candidat: Roxana-Denisa Vîrlan**

**Coordonator științific: S.L. dr. ing. Alexandru Iovanovici  
Drd. ing. Dacian Avrămoni**

Iunie 2022

## REZUMAT

Lucrarea propune proiectarea, implementarea și testarea unui sistem *embadded* de tip dozator inteligent de medicamente cu detecția automată a ingestiei pilulei, pentru urmărirea complianței pacientului la administrarea tratamentului medicamentos pe cale orală.

Acest sistem este format din: un dozator inteligent de medicamente și un dispozitiv sub formă de brățară utilizat pentru detecția gesturilor care are la bază o rețea neuronală. De asemenea, acest sistem interacționează cu o aplicație mobilă prin intermediul căreia este setată medicația utilizatorului și care notifică persoana de contact a celui care utilizează acest sistem în cazul în care tratamentul nu a fost urmat.

Dozatorul inteligent de medicamente are funcționalități precum: setarea alarmelor și eliberarea automată a medicamentelor în funcție de medicația stabilită prin intermediul aplicației mobile, notificarea utilizatorului atunci când medicamentele din recipiente sunt epuizate și reumplerea recipientelor prin intermediul unei interfețe grafice intuitive. De asemenea, dozatorul, prin intermediul unei camere de fotografiat capturează o imagine cu recipientul în care sunt eliberate pasilele pentru a se asigura că pastilele au fost luate, după ce s-a realizat procesul de ingestie a medicamentului.

Procesul de ingestie este asigurat prin intermediul unui dispozitiv de detecție ce folosește un algoritm de învățare automată de tip Machine Learning. Datele utilizate pentru antrenarea algoritmului au fost colectate de la un număr de 14 persoane, cu vârste diferite.

Dispozitivul de detecție, sub formă de brățară ce este atașată la încheietura mâinii utilizatorului, are ca scop detecția mișcării de ducere a mâinii la gură în procesul de ingestie a medicamentului, după ce acesta a fost eliberat din dozatorul inteligent.

Sistemul este ușor de utilizat ceea ce reprezintă un aspect important deoarece se adresează în special persoanelor vârstnice sau cele cu dificultăți de memorie.

Acest sistem a fost testat, iar în urma testelor s-a constatat că funcționează corespunzător .

## CUPRINS

1. INTRODUCERE .....	4
1.1 Context .....	4
1.2 Descrierea proiectului .....	4
1.3 Structura lucrării .....	5
2. SPECIFICAȚIILE SISTEMULUI .....	7
2.1 Analiza domeniului.....	7
2.1.1 Cutii de medicamente asemănătoare .....	7
2.1.2 Alți algoritmi de detecție a gesturilor .....	11
2.2 Lista specificațiilor sistemului dezvoltat .....	12
3. ARHITECTURA SISTEMULUI .....	14
4. IMPLEMENTAREA SISTEMULUI .....	16
4.1 Dispozitiv pentru detecția gesturilor .....	16
4.1.1 Colectarea seturilor de date .....	17
4.1.2 Crearea algoritmului de detecție a gesturilor.....	19
4.1.3 Programarea sistemului de inferență al rețelei neuronale.....	23
4.2 Dozator inteligent de medicamente EZ Pills .....	23
4.2.1 Resurse hardware .....	24
4.2.2 Conectarea componentelor.....	27
4.2.3 Software <i>embedded</i> .....	29
5. TESTAREA SISTEMULUI .....	34
5.1 Testarea sistemului de detecție a gesturilor .....	34
5.2 Testarea dozatorului inteligent de medicamente .....	36
5.3 Testarea întregului sistem .....	37
6. CONCLUZII.....	38
6.1 Realizări.....	38
6.2 Dezvoltări ulterioare.....	39
BIBLIOGRAFIE .....	40
Anexe – Declarație de autenticitate a lucrării de finalizare a studiilor.....	43

## 1. INTRODUCERE

### 1.1 Context

Pornind de la citatul:

„Să îți dorești să fii sănătos este o parte din sănătate.”

care aparține filosofului Seneca și dorindu-mi să rezolv o problemă din sistemul medical, care ar putea menține sau chiar îmbunătăți starea de sănătate a oamenilor, am decis să mă documentez despre situația ce există în momentul de față în ceea ce privește tratamentul ambulatoriu al pacienților din România.

Conform studiului *State of Health in the EU*, în România, rata mortalității evitabile prin prevenție și a mortalității prin cauze tratabile sunt foarte ridicate [1].

Majoritatea pacienților se lovesc de următoarele probleme: au luat prea multe medicamente sau au uitat orele la care ar trebui să ia medicamentele prescrise de medic, confundă numele medicamentelor sau, din cauza faptului că au rețete de medicamente complexe, nu pot urma un tratament corect. Problema devine cu atât mai gravă cu cât persoanele care urmează un anumit tratament pot suferi de boli care le afectează sistemul nervos (cea mai cunoscută fiind boala Alzheimer). Conform studiului realizat de *Global Burden of Disease* [2], din ce în ce mai multe persoane se confruntă cu boli ale sistemului nervos manifestate prin pierderea memoriei, iar numărul acestora este în continuă creștere. Administrarea tratamentului pentru astfel de persoane este absolut necesară deoarece, în lipsa unui plan de tratament corect, se ajunge la degradarea profundă și iremediabilă a sănătății.

Lipsa administrării medicamentelor prescrise de medici nu afectează doar sănătatea pacienților în cauză, ci întreg sistemul medical deoarece starea de sănătate se poate agrava, ajungându-se la spitalizare. Conform studiului realizat de *Personal Health* [3], neurmarea unui plan de tratament corect a cauzat în anul 2017, în Statele Unite ale Americii, într-o perioadă de epidemie, aproximativ 125000 de decese și un cost total de aproximativ 289 miliarde de dolari pentru sistemul medical deoarece rata de spitalizare a crescut cu 10%.

Soluția găsită pentru situațiile dificile cu care se confruntă pacienții în ceea ce privește tratamentul ambulatoriu este respectarea unui plan de tratament. Acest lucru poate fi realizat prin notificarea pacienților și dozarea automată a pastilelor în momentul în care tratamentul trebuie administrat. Pentru a veni în ajutorul persoanelor care au probleme cu memoria, sistemul verifică dacă tratamentul a fost într-adevăr urmat prin intermediul unui dispozitiv pentru detecția ingestiei de pastile.

### 1.2 Descrierea proiectului

Așa cum am descris anterior, ideea principală pe care se bazează sistemul dezvoltat a fost găsirea unei soluții pentru a facilita tratamentul ambulatoriu al pacienților, având în

vedere problemele enunțate în subcapitolul anterior. Utilizatorii sistemului vor urma un plan de tratament corect care le-ar putea menține sau îmbunătăți starea de sănătate.

Datorită progresului permanent al tehnologiilor *hardware* și *software*, am reușit să creez un sistem format dintr-un dozator inteligent de medicamente și un dispozitiv de detecție a gesturilor. Acest sistem ușurează administrarea tratamentului utilizatorilor prin dozarea automată a pastilelor după un plan de tratament stabilit de medic și prin verificarea ingestiei pastilei. Astfel, tratamentul poate fi monitorizat, chiar și de la distanță, prin intermediul unei aplicații mobile cu care interacționează sistemul dezvoltat.

Odată setat planul de tratament, dozatorul de medicamente, la orele stabilite pentru administrarea tratamentului, va seta o alarmă și va elibera automat pastila. Astfel, pacienții ce utilizează sistemul pe care l-am dezvoltat nu vor mai trebui să își facă griji cu privire la momentele din zi în care trebuie să își ia tratamentul. Adăugarea pastilelor în acest dozator se face simplu, prin intermediul unei interfețe grafice de unde se selectează ce medicament se va adăuga și numărul de pastile. De asemenea, când recipientele sunt aproape goale, se va primi o notificare care anunță acest lucru.

Dispozitivul de detecție a gesturilor este sub forma unui ceas atașat la încheietura mâinii. Acest dispozitiv are rol în detectarea momentului în care pacientul a ridicat mâna pentru a-și lua pastila după ce aceasta a fost eliberată din dozator. Pentru detecția gesturilor s-a folosit un algoritm de Machine Learning. Printr-o conexiune Bluetooth, acesta transmite aplicației mobile cu care interacționează dacă pastila a fost luată sau nu.

Dacă tratamentul s-a considerat administrat, dozatorul capturează o fotografie cu recipientul în care sunt eliberate pastilele. Această fotografie va fi încărcată într-o bază de date, pentru cazul în care o persoană de contact a utilizatorului dorește să verifice dacă a luat sau nu pastila din acel recipient.

Utilizarea acestui sistem este facilă, ținându-se cont că utilizatorii țintă sunt persoanele vârstnice sau cele cu probleme cu memoria.

Astfel, sistemul dezvoltat de mine își propune să implementeze următoarele funcționalități:

- Setarea alarmelor
- Dozarea automată a pastilelor
- Reumplerea stocului de medicamente printr-o interfață grafică intuitivă
- Detecția ingestiei pilulei cu ajutorul unui algoritm de Machine Learning
- Capturarea unei fotografii pentru a vedea dacă pastilele au fost luate
- Realizarea legăturii dintre dozatorul inteligent, brățară și aplicație mobilă

### 1.3 Structura lucrării

Lucrarea este structurată în 6 capitole, după cum urmează: *Introducere*, *Specificațiile sistemului*, *Arhitectura sistemului*, *Implementarea sistemului*, *Testarea sistemului* și *Concluzii*.

În capitolul *Introducere* s-au prezentat aspecte generale cu privire la domeniul de studiu abordat, cât și descrierea lucrării și obiectivele propuse.

În capitolul *Specificațiile sistemului* s-au prezentat aspecte teoretice despre necesitatea specificațiilor în dezvoltarea software, analiza domeniului, prezentând sisteme asemănătoare de dozare a medicamentelor și algoritmi de detecție a gesturilor, dar și specificațiile sistemului dezvoltat de mine.

În capitolul *Arhitectura sistemului* s-au prezentat arhitectura generală a sistemului și cum au fost rezolvate specificațiile sistemului de la capitolul anterior.

În capitolul *Implementarea sistemului* s-au prezentat pe larg etapele necesare realizării proiectului din punct de vedere practic (resursele software, resursele hardware, dar și conectarea componentelor).

În capitolul *Testarea sistemului* s-au prezentat testele efectuate asupra dispozitivului de detecție a gesturilor, dozatorului inteligent de medicamente, dar și asupra întregului sistem.

În capitolul *Concluzii* s-au prezentat realizările, dar și posibilele dezvoltări ulterioare.

## 2. SPECIFICAȚIILE SISTEMULUI

Specificațiile unui sistem reprezintă descrierea a ceea ce ar trebui să facă sistemul, mai exact funcționalitățile și serviciile pe care acesta le oferă și constrângerile asupra funcționării acestuia [4].

Specificațiile unui sistem se pot clasifica în specificații funcționale și specificații non-funcționale [4]:

- Specificațiile funcționale se referă la funcționalitățile pe care sistemul ar trebui să le ofere și cum ar trebui să reacționeze sistemul la anumite situații. De asemenea, specificațiile funcționale precizează în mod explicit ce nu ar trebui să facă sistemul dezvoltat.
- Specificațiile non-funcționale reprezintă constrângerile asupra serviciilor sau funcționalităților oferite de sistem.

### 2.1 Analiza domeniului

#### 2.1.1 Cutii de medicamente asemănătoare

Cutiile de medicamente convenționale nu mai sunt o opțiune atunci când vine vorba de persoanele care au probleme cu memoria sau care au rețete de medicamente destul de complexe. Din acest motiv, cutiile de medicamente inteligente pot fi soluția problemelor în astfel de situații.

Pentru a evidenția caracteristicile sistemului dezvoltat de mine, numit EZ Pills, am ales să îl compar cu alte sisteme cu funcționalități asemănătoare. Conform site-ului „*Tech-enhanced Life*”, *MedaCube*, *MedMinder* și *Hero* [5] sunt în topul preferințelor utilizatorilor în materie de cutii inteligente de medicamente.

Caracteristicile MedaCube (Figura 2.1) [6]:

- Ecran tactil prin intermediul căruia se pot seta programul și dozajul medicamentelor
- 16 tipuri de pastile diferite
- Necesarul de pastile pentru până la 90 de zile
- Alerte sonore și vizuale pentru a lua medicamentele
- Transmite notificări persoanelor care se ocupă de pacienți dacă aceștia nu și-au luat medicamentele
- Notificări pentru reumplerea recipientelor cu medicamente atunci când acestea s-au golit
- Meniul supraveghetorului este protejat prin parolă
- Fotografiera recipientelor în care se află medicamentele înainte și după administrarea dozei și încărcarea acestei imagini, astfel încât supraveghetorii să vadă dacă sistemul funcționează corect

- Conexiune la internet fie prin conexiunea celulară incorporată, fie prin WiFi
- Cameră care citește codurile de bare de pe recipientele/cutiile medicamentelor
- Senzor care detectează dacă cutia a fost mutată
- Baterie de rezervă până la 24h
- Dimensiuni mari, ceea ce îl face greu de transportat
- Nu acceptă capsule cu gel lichid sau cele care se oxidează în contact cu aerul
- Preț: 1599.00\$

#### Caracteristicile MedMinder (Figura 2.2) [7]:

- Proiectat ca o cutie convențională de medicamente, ceea ce presupune setarea manuală a medicației în celule
- Alerte sonore și luminoase
- 28 de compartimente ( o doză pe zi timp de 28 de zile sau 4 doze pe zi timp de o săptămână)
- Orice tip de medicament
- Nu are capacitate de recunoaștere a codului de bare pentru recipientele medicamentelor
- Transmite notificări persoanelor care se ocupă de pacienți dacă aceștia nu și-au luat medicamentele
- Interfață web care se conectează la cutia de medicamente
- Flexibilitate: medicamentele pot fi luate înainte sau după program
- Senzor pentru detecție dacă medicamentul a fost luat sau nu din compartiment
- Conexiune GSM incorporată
- Baterie de rezervă de până la 48h
- Preț: 45\$

#### Caracteristicile Hero (Figura 2.3) [8], [9]:

- 10 tipuri de pastile diferite
- Necesarul de pastile pentru până la 90 de zile
- Orice tip de medicamente, în afară de capsule cu gel lichid sau cele ce se oxidează în contact cu aerul
- Notificări sonore și luminoase
- Setarea programului și dozajul medicamentelor se face prin intermediul unei aplicații mobile/web
- Transmite notificări persoanelor care se ocupă de pacienți dacă aceștia nu și-au luat medicamentele
- Interfața cu utilizatorul este asigurată de butoane și ecran



- Nu are capacitate de recunoaștere a codului de bare pentru recipiente medicamentelor
- Dispozitivul dispune de tehnologie care comandă automat medicamentele care s-au epuizat
- Aplicație mobilă
- Conexiune la internet prin WiFi
- Dimensiuni mari, ceea ce îl face greu de transportat
- Fără baterie de rezervă
- Preț: 99,99\$ + 29.99\$ taxă lunară pentru membri



Figura 2.1: : Dozatorul automat de pastile „MedaCube” [10]



Figura 2.2: Cutia de medicamente „MedMinder” [11]



Figura 2.3: Dozatorul de medicamente „Hero” [12]

Tabelul 2.1 compară dozatorul „EZ Pills” cu trei sisteme aflate în topul preferințele utilizatorilor în materie de cutii/dozatoare inteligente de medicamente. Se poate observa cu ușurință ca aproape orice funcționalitate ce se regăsește în acestea este disponibilă și utilizatorilor sistemului dezvoltat de mine, exceptând scanarea codului de bare pentru recunoașterea medicamentului.

Ceea ce face diferența între „EZ Pills” și celelalte dozatoare/cutii de medicamente este atât dovada foto că pastila a fost luată din paharul în care aceasta a fost eliberată, dar mai ales modul în care sistemul este interconectat cu aplicația mobilă și dispozitivul de detecție a gesturilor atașat la încheietura mâinii. Pacientul beneficiază atât de un dozator inteligent de medicamente, cât și de verificarea suplimentară cu ajutorul dispozitivului pe care îl poartă că medicamentul a fost într-adevăr luat.

*Tabelul 2.1: Analiză comparativă între sistemul dezvoltat de mine și alte trei sisteme asemănătoare*

<b>Caracteristici și funcționalități</b>	<b>Dozatorul automat MedaCube</b>	<b>Cutia de medicamente MedMinder</b>	<b>Dozatorul Hero</b>	<b>Dozatorul EZ Pills</b>
x tipuri de pastile	16	-	10	4
Tip de medicamente	Fără capsule cu gel lichid sau cele ce se oxidează în contact cu aerul	Orice tip de medicament	Fără capsule cu gel lichid sau cele ce se oxidează în contact cu aerul	Fără capsule cu gel lichid sau cele ce se oxidează în contact cu aerul
Notificări pentru a lua medicamentul	Sonore și vizuale	Sonore și vizuale	Sonore și vizuale	Sonore și vizuale
Notificări pentru supraveghetori	Da	Da	Da	Da
Notificări pentru reumplere la epuizarea pastilelor	Da	Nu	Da	Da
Interfață cu utilizatorul	Ecran tactil	-	Ecran și butoane	Ecran tactil
Setarea medicației	Ecran tactil sau la distanță prin aplicație	Manual, ca la o cutie convențională	Prin intermediul aplicației	Prin intermediul aplicației
Recunoașterea codului de bare	Da	Nu	Nu	Nu
Aplicație mobilă/web	Aplicație mobilă/web	Aplicație web	Aplicație mobilă/web	Aplicație mobilă
Baterie de rezervă	Da, până la 24h	Da, până la 48h	Nu	Da, până la 24h
Conexiune la internet	WiFi și conexiune celulară incorporată	Conexiune celulară incorporată	WiFi	WiFi

Fotografierea paharului cu medicamente	Nu	Nu	Nu	Da
Preț	1599.00\$	45\$	99,99\$ + 29,99\$ taxă de membru	-

### 2.1.2 Alți algoritmi de detecție a gesturilor

Pentru detecția ingestiei medicamentelor am optat pentru un algoritm de învățare automată de tip Machine Learning. Deoarece se dorește clasificarea mișcărilor mâinii, dacă a dus sau nu mâna la gură să își ia medicamentul, am considerat potrivită o rețea neuronală complet conectată care să folosească un algoritm supervizat de clasificare.

Pentru o analiză comparativă a algoritmilor de detecție a gesturilor, am consultat multiple surse ale literaturii de specialitate. În domeniul recunoașterii gesturilor se diferențiază două abordări:

- Vision-Based Recognition (VBR) care efectuează recunoașterea gesturilor din imaginile capturate de o cameră;
- Sensor-Based Recognition (SBR) presupune recunoașterea gesturilor utilizând senzori precum: accelerometru, giroscop, RFID (Radio-Frequency Identification);

Cea de-a doua abordare a fost de interes pentru sistemul dezvoltat de mine.

În articolul [13], autorii au propus un algoritm de detecție a gesturilor care utilizează o rețea neuronală RCE (restricted column energy) unde se face înlocuirea calculului metric cu distanța obținută prin tehnica DTW(dynamic time warping). Rețeaua neuronală RCE este formată din trei straturi (layers): strat de intrare (input layer), strat ascuns (hidden layer) și strat de ieșire (output layer), iar neuronii sunt activați doar atunci când este necesar. Precizia pentru recunoașterea gesturilor a fost de 98,6%.

În articolul [14], autorii au propus algoritmi de recunoaștere eficientă a gesturilor care utilizează o rețea neuronală de tip feedforward. Algoritmii propuși se bazează pe PairNet. Aceștia au precizie ridicată de clasificare a gesturilor și complexitate de calcul scăzută. Un număr de 11 clase de gesturi au fost studiate și folosite pentru controlul de la distanță a electrocasnicelor inteligente (smart-TV, aer condiționat, music player). Precizia de recunoaștere a gesturilor obținută în urma a 3404 de teste a fost de 92.36%.

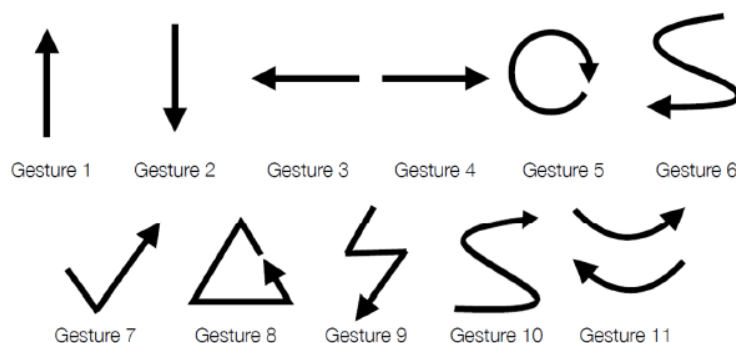


Figura 2.4: Cele 11 clase de gesturi considerate pentru controlul de la distanță a electrocasnicelor inteligente [14]

## 2.2 Lista specificațiilor sistemului dezvoltat

Sistemul are ca scop facilitarea tratamentului ambulatoriu al pacienților. Pentru a realiza un astfel de sistem, acesta trebuie să aibă următoarele funcționalități:

- La orele stabilite printr-un plan de tratament, utilizatorul ar trebui să fie notificat pentru a-și aminti să-și ia pastilele
- Planul de tratament ar trebui stocat într-o bază de date pentru a putea fi gestionat
- Dozarea medicamentelor ar trebui să fie automată
- Sistemul care are rol în dozarea medicamentelor ar trebui să aibă o interfață facilă pentru reumplerea stocului de pastile
- Sistemul să poată interacționa cu o aplicație mobilă cu rol în setarea medicației utilizatorului
- Legătura dintre dispozitivul de dozare al pastilelor și aplicație să poată fi făcută printr-un cod de identificare unic
- În cazul unor erori de funcționare, utilizatorul sau persoanele de contact ale acestuia ar trebui să fie notificate
- Pentru un plus de siguranță că tratamentul a fost administrat este necesar un dispozitiv pentru detecția ingestiei

Sistemul ar putea avea unele constrângeri precum:

- Dispozitivul pentru dozare a pastilelor ar putea avea dimensiuni mari, ceea ce îl face greu de transportat.

- Dispozitivele care utilizează baza de date ar trebui să aibă acces continuu la Internet

### 3. ARHITECTURA SISTEMULUI

Acest capitol cuprinde schema arhitecturii generale a sistemului: prezentarea componentelor și interconectarea acestora.

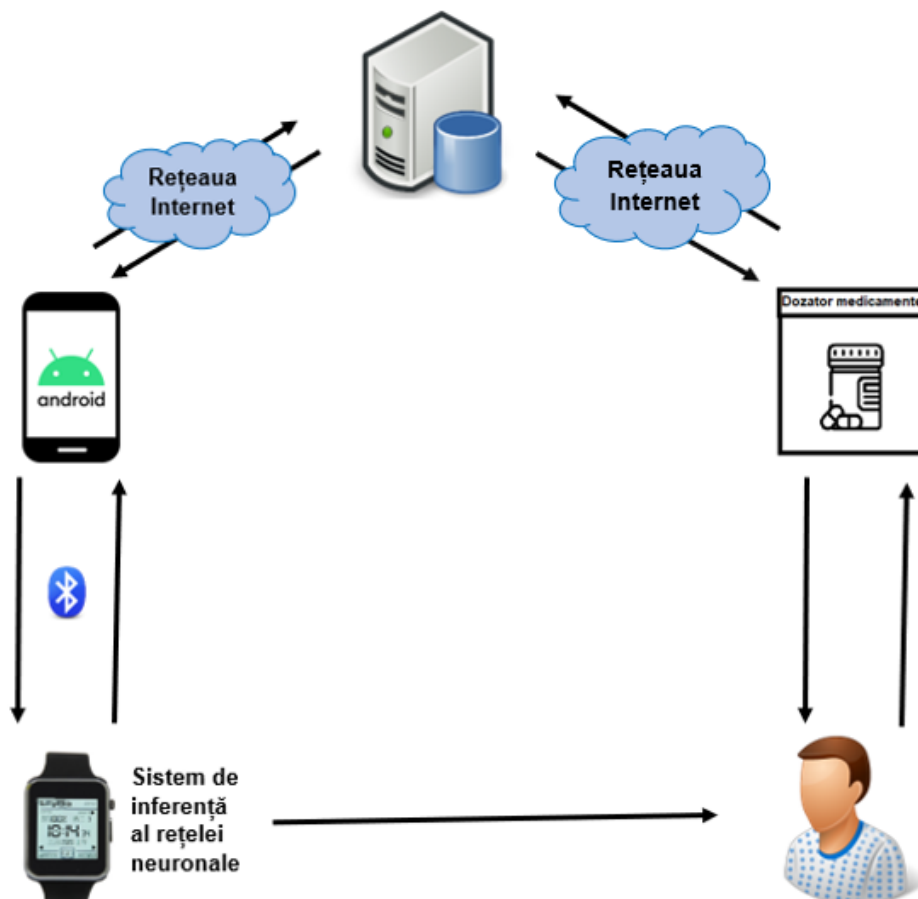


Figura 3.1: Schema generală a sistemului

Sistemul este format din 4 subsisteme: o aplicație mobilă Android, un dozator inteligent de medicamente, o bază de date și un dispozitiv de detecție a gesturilor sub formă de brățară.

Baza de date este utilizată pentru a putea susține funcționalitatea aplicației mobile și a dozatorului inteligent de medicamente.

Legătura dintre aplicația mobilă și dozatorul inteligent de medicamente se realizează prin introducerea unui cod unic al cutiei în aplicație. În baza de date se poate vizualiza secțiunea destinată fiecărei cutii cu care este asociată aplicația mobilă.

Aplicația Android are rolul de a seta medicația utilizatorului și de a-l notifica la orele la care ar trebui să își ia tratamentul. Conexiunea dintre dispozitiv și baza de date se realizează prin intermediul rețelei Internet. Informații precum: ora următoarei alarme programate, numele medicamentelor și compartimentul acestora din dozator sunt stocate în baza de date prin intermediul aplicației.

Pentru stocarea și sincronizarea datelor, dozatorul de medicamente interacționează cu baza de date, printr-o conexiune WiFi, astfel: actualizează numărul de medicamente din fiecare compartiment, odată cu adăugarea unor noi medicamente sau după ce o pastilă a fost eliberată, notifică de fiecare dată când un medicament este eliberat, trimite dovadă o

fotografie cu recipientul în care sunt eliberate medicamentele pentru ca supraveghetorul pacientului să verifice dacă medicamentele au fost într-adevăr luate. Dozatorul primește de la baza de date informații precum ora următoarei alarme, numele medicamentului următor ce trebuie luat și compartimentul în care acesta se află.

Reumplerea recipientelor dozatorului se face prin intermediul unei interfețe grafice afișată pe ecranul dispozitivului. Odată selectat medicamentul pentru care se dorește această acțiune și numărul de pilule ce se vor adăuga, se va indica recipientul care conține acel medicament. De asemenea, tot prin intermediul acestei interfețe grafice, utilizatorul poate vedea câte medicamente au mai rămas în fiecare recipient.

Dispozitivul de detecție a gesturilor comunică cu aplicația mobilă prin standardul Bluetooth 4.2. Pe baza algoritmului de detecție a gesturilor, sistemul verifică dacă s-a efectuat mișcarea de ducere a mâinii la gură după ce pastila a fost eliberată, iar în caz afirmativ, acesta transmite aplicației că medicamentul a fost luat. Dacă nu s-a detectat acea mișcare în intervalul alocat luării pastilei, se consideră că medicamentul nu a fost luat, marcându-se acest lucru în aplicație și oprindu-se alarma.

Procesul de administrare a medicamentului este ilustrat în Figura 3.2:

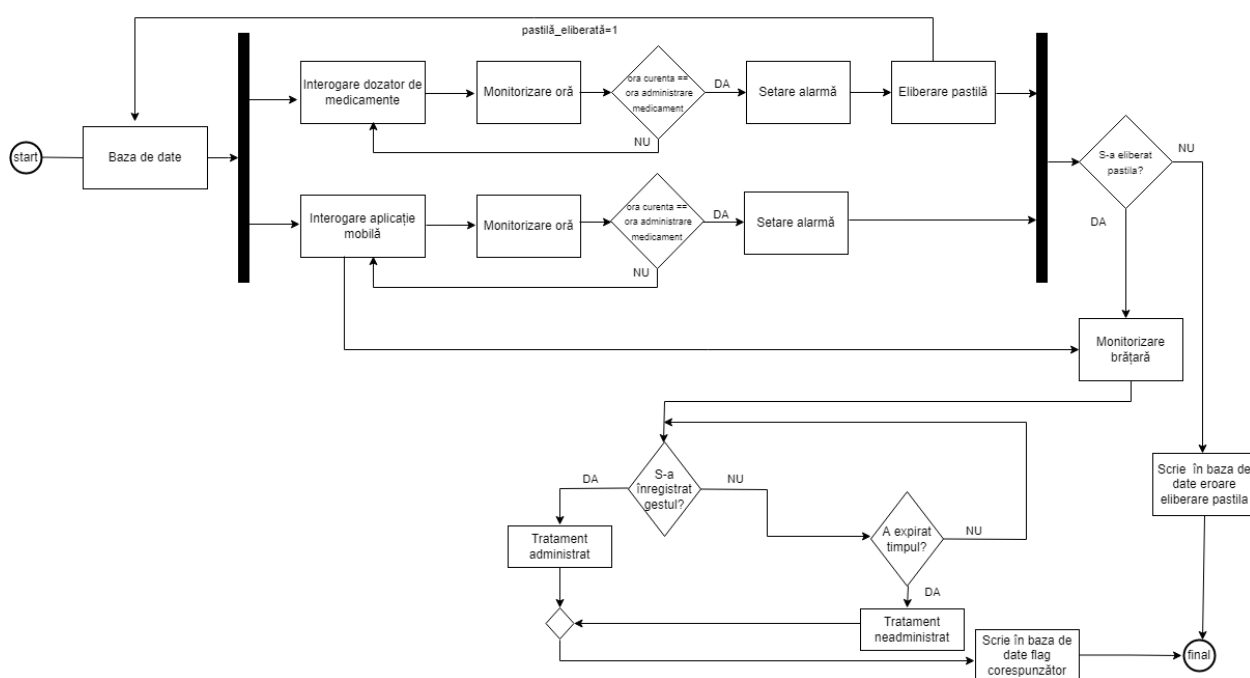


Figura 3.2: Diagrama de evenimente a sistemului pentru administrarea medicamentelor

Conform Figurii 3.2, pacientul este notificat de către dozatorul inteligent, printr-o alarmă, în momentul în care trebuie să ia un anumit medicament. După începerea alarmei, dacă sunt suficiente medicamente în compartiment, dozatorul eliberează câte o pilulă și așteaptă să primească semnalul că tratamentul a fost administrat. Alarma se va opri în două situații: fie pacientul și-a luat medicamentul, fie a trecut timpul acordat luării medicamentului. Există o situație specială în care se poate opri alarma și anume, dacă în urma încercărilor de eliberare a medicamentelor, nu a fost eliberat niciun medicament. În acest caz se consideră o eroare de funcționare a dozatorului, iar alarma se va opri, marcându-se în baza de date acest lucru.

## 4. IMPLEMENTAREA SISTEMULUI

Acest capitol va descrie pe larg etapele necesare realizării proiectului din punct de vedere practic. Se vor prezenta resursele hardware și resursele software, dar și modul de conectare al componentelor.

### 4.1 Dispozitiv pentru detecția gesturilor

Pentru partea de recunoaștere a gestului de ducere a mâinii la gură în procesul de ingestie a medicamentului am optat pentru un ceas open-source interactiv numit *LilyGo TTGO T-Watch 2020* care poate fi ușor de programat și de utilizat în viața de zi cu zi.

LilyGo TTGO T-Watch are la bază un controller de tip ESP32 și componente adiționale precum: accelerometru cu trei axe BMA423, transmițător infraroșu, modul RTC, amplificator audio și modul GPS. Dispozitivul dispune de un ecran tactil capacitiv cu rezoluția de 1,54 inch, o baterie cu litiu de 380mAh și un buton lateral cu funcție de pornire/oprire. Acesta poate fi programat în Arduino, ESP-IDF, Lua, Scratch sau MicroPython [15].



Figura 4.1: Ceas LilyGO TTGO T-Watch 2020 [15]

Componenta de bază a acestui ceas îl constituie microcontrollerul **ESP32**. Acesta are integrat un modul Wi-Fi, care se poate folosi atât ca stație Wi-Fi pentru conectarea la internet, cât și pentru a crea un punct de acces la care se pot conecta alte dispozitive. Pe lângă Wi-Fi, acesta oferă tehnologie Bluetooth 4.2, dar și conectivitate Bluetooth low-energy [16]. Acesta dispune de o memorie FLASH de 16MB, memorie SRAM de 520kB și memorie PSRAM de 8MB.

**BMA423** este un senzor de dimensiuni foarte mici, utilizat pentru măsurarea accelerației, având o rezoluție digitală de 12 biți. Comunicarea dintre procesor și BMA423 se poate realiza prin două protocoale de comunicare serială: I2C (Inter-Integrated Circuit)



sau SPI (Serial Peripheral Interface). Conform fișei cu specificații, acesta dispune de o coadă de tipul „First In, First Out” (FIFO) de 1kB care reduce traficul pe interfața serială, dar și consumul de energie. Acest accelerometru permite un consum foarte mic de curent, aproximativ 13  $\mu$ A în modul de funcționare cu consum redus de energie și 150  $\mu$ A în modul de performanță ridicată, făcându-l ideal pentru dispozitivele de tip ceasuri inteligente pentru monitorizarea activităților zilnice.

Pentru a putea detecta gestul de ducere a mâinii la gură în procesul de ingestie a medicamentelor am folosit un algoritm de Machine Learning.

*Machine Learning*-ul este o ramură a inteligenței artificiale care se bazează pe utilizarea datelor și algoritmilor pentru a imita modul de gândire al oamenilor. Un algoritm de Machine Learning are trei părți principale: un proces de decizie (algoritmii sunt utilizați fie pentru a face predicții, fie pentru clasificare), o funcție de estimare a erorii (pentru a evalua acuratețea modelului) și un proces de optimizare a modelului (pentru ajustarea ponderilor până la atingerea unei valori optime) [17].

Algoritmii de Machine Learning se împart în trei categorii, în funcție de stilul de învățare:

- *Supervised learning*, unde modelul este antrenat cu ajutorul unor date care au un rezultatul predefinit
- *Unsupervised learning*, unde modelul lucrează singur, fără intervenția umană, pentru a decoperi tipare și informații nedescoperite anterior. Cu alte cuvinte, seturile de date nu au un rezultat specific.
- *Semi-supervised learning*, combină datele *labeled*, cu cele *unlabeled*, fiind o combinație între învățarea supervizată și cea nesupervizată

Dezvoltarea unui algoritm de Machine Learning implică: colectarea seturilor de date, pregătirea datelor, alegerea unui model și antrenarea și evaluarea acestuia.

Pentru sistemul dezvoltat, deoarece se dorește clasificarea mișcărilor mâinii, dacă a dus sau nu mâna la gură să își ia medicamentul, am considerat potrivită o rețea neuronală complet conectată care să folosească un algoritm supervizat de clasificare.

Pentru a putea antrena și testa algoritmul de detecție a gestului am avut nevoie de seturi de date. Pentru o detecție precisă care să funcționeze indiferent de persoana care utilizează dispozitivul, datele trebuie să fie colectate de la persoane diferite.

#### 4.1.1 Colectarea seturilor de date

Pentru colectarea seturilor de date am creat o interfață cu utilizatorul compusă dintr-o interfață grafică și o interfață haptică (vibrație), care permite gestionarea procesului de măsurare a tuturor parametrilor necesari. Utilizatorul decide inițierea procesului de măsurare prin atingerea ecranului tactil, după care se permite o perioadă de așteptare de 5 secunde, timp în care utilizatorul își poziționează mâna în poziția neutră, urmată de semnalarea începerii acțiunii de măsurare. În perioada măsurării (1.5 secunde), utilizatorul trebuie să efectueze gesturile asociate ingestiei unei pastile (pentru colectarea datelor am

folosit produse de tip bomboane), iar la sfârșitul colectării datelor se semnalează de asemenea prin vibrație.

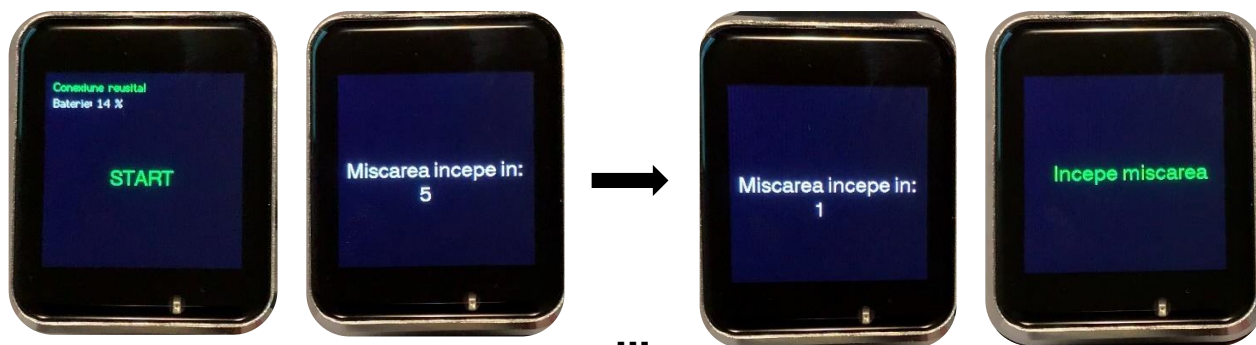


Figura 4.2: Interfața grafică pentru colectarea seturilor de date

Interfața grafică afișează dacă sistemul s-a conectat sau nu la internet („Conexiune reușită!” / „Conexiune eșuată”) și procentajul bateriei. Conexiunea la internet este esențială pentru procesul de colectare a datelor deoarece datele primite de la accelerometru sunt preluate de la dispozitiv folosind *WebSocket*-uri.

Un *WebSocket* este o conexiune între client și server care permite comunicarea bidirecțională folosind o conexiune TCP. Clientul stabilește o conexiune *WebSocket* cu serverul printr-un proces cunoscut sub numele de *handshaking*. Procesul începe cu o cerere/răspuns HTTP, permițând serverelor să gestioneze conexiunile HTTP, precum și conexiunile *WebSocket* pe același port. După ce s-a stabilit conexiunea, clientul și serverul pot trimite date *WebSocket* în modul full duplex [18]. *WebSocket*-urile se folosesc în special în dezvoltarea aplicațiilor *real-time* deoarece transmisia datelor este rapidă, sporindu-se astfel performanța acestora.

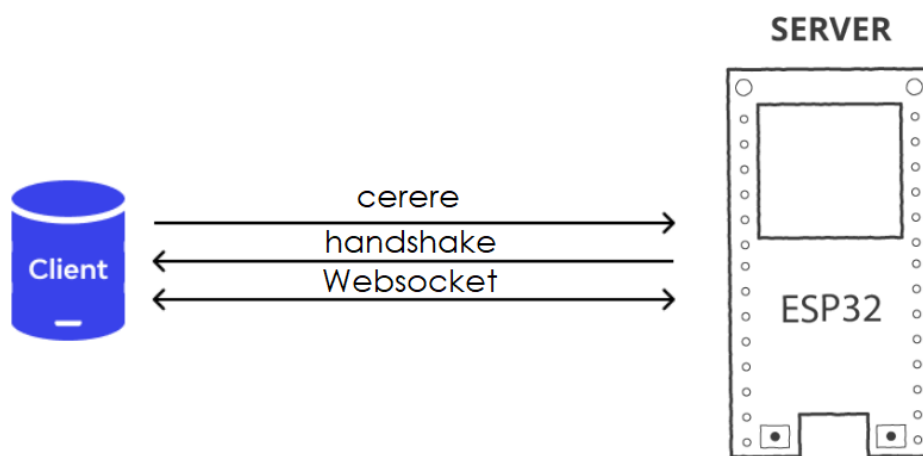


Figura 4.3: Conexiunea *WebSocket*

### Secvența de cod pentru conectarea la internet:

```
WiFi.mode(WIFI_STA);  
WiFi.begin(ssid, password);  
while (WiFi.waitForConnectResult() != WL_CONNECTED) {  
    tft->setTextColor(TFT_RED, TFT_BLACK);  
    tft->drawString("Conexiune esuata! Reconectare...", 10, 10, 2);  
    delay(4000);  
    ESP.restart();  
}
```

Pentru transmiterea datelor către server, am folosit următoarele biblioteci:

- WebSocketsServer.h
- ESPAsyncWebServer.h

Datele au fost trimise sub forma  $x_1, y_1, z_1; \dots, x_{150}, y_{150}, z_{150};$ .

Cu ajutorul unui script Python, datele au fost puse într-un fișier .xlsx. În acest fișier, există trei coloane (corespunzătoare accelerațiilor pe cele trei axe – aX, aY și aZ) și 150 de rânduri pentru fiecare măsurare (corespunzătoare numărului de eșantioane preluate), noua măsurare fiind marcată de un rând liber.

Secvența de cod corespunzătoare pentru preluarea datelor și punerea acestora într-un fișier .xlsx este următoarea:

```
while 1:  
    result = ws.recv()  
    if len(result)>100:  
        splitted_elements = result.split(';')  
        current_row = sheet.max_row + 1  
        for i in range(0, len(splitted_elements)-1):  
            #am masuratoare ax, ay, az  
            aX, aY, aZ = splitted_elements[i].split(',')  
            sheet.cell(current_row, 1).value = aX  
            sheet.cell(current_row, 2).value = aY  
            sheet.cell(current_row, 3).value = aZ  
            current_row = current_row + 1  
        current_row = current_row + 1  
        wb.save(data_path)  
        result = ""
```

#### 4.1.2 Crearea algoritmului de detecție a gesturilor

Pentru implementarea algoritmului de Machine Learning utilizat pentru detecția gesturilor am folosit limbajul de programare Python.

Conform unei statistici realizate în luna Octombrie 2021, Python este cel mai popular limbaj de programare din lume [19]. Python este un limbaj de programare dinamic și flexibil,

iar spre deosebire de celelalte limbaje care folosesc semne de punctuație și paranteze pentru a marca finalizarea unei comenzi, acesta folosește spații albe/linii noi. Dinamicitatea limbajului este dată de faptul că nu se specifică tipul variabilelor în momentul declarării lor, ci tipul este determinat ulterior de interpretor după valorile pe care le conțin sau după operațiile în care sunt implicate. Codul este interpretat în timpul rulării de interpretor (runtime), deci programul nu trebuie compilat înainte de a fi executat.

Fără urmă de îndoială, în domeniul Machine Learning și Data Science, Python a reușit să se impună datorită numeroaselor biblioteci de matematică și statistică, extensibilității sale, dar și ușurinței de utilizare.

În cadrul proiectului, am utilizat diferite biblioteci: tensorflow, matplotlib (pentru a reprezenta grafic datele), numpy (pentru prelucrarea matricelor) și pandas (pentru manipularea și analiza datelor).

Biblioteca Tensorflow este o bibliotecă open-source destinată calculelor numerice de înaltă performanță pentru acele tipuri de operații de care e nevoie în procesul de învățare automată folosind rețelele neuronale.

Interfața de programare (Application Programming Interface – API) care rulează pe platforma de învățare automată Tensorflow este Keras [20]. În Keras, datele sunt organizate în straturi și modele. Procesele precum antrenarea, testarea și evaluarea modelelor se realizează simplu, prin apelarea funcțiilor: fit, evaluate și predict.

Algoritmul de detecție a gesturilor a fost realizat în Google Colaboratory. Acesta este o mediu de dezvoltare gratuit, bazat pe cloud, unde se poate scrie, vizualiza și executa codul în celule și permite antrenarea modelelor pentru învățare automată.

Pentru dezvoltarea acestui sistem de detecție, am considerat potrivită o rețea neuronală complet conectată care să folosească un algoritm supervizat de clasificare. O rețea neuronală reprezintă o colecție de neuroni interconectați care funcționează împreună pentru realizarea unui scop comun. Rețeaua neuronală are cu un set de ponderi  $w$  care transformă datele de intrare în straturi ascunse, iar aceste ponderi sunt optimizate în procesul de antrenare. Așadar, fiecare neuron din straturile ascunse înmulțește intrarea cu o pondere, numită *weight*, și o însumează cu o valoare numită *bias*.

$$ieșirea = \sum(w * intrare) + bias \quad (1)$$

Dacă intrarea este de forma  $x_1, x_2, \dots, x_n$ , iar ponderea de forma  $w_1, w_2, \dots, w_n$ , atunci suma ponderată este calculată astfel [21]:

$$x_1 * w_1 + x_2 * w_2 + \dots + x_n * w_n \quad (2)$$

Pentru algoritmul de detecție a gesturilor, modelul este alcătuit dintr-un strat de intrare cu 50 de neuroni, un strat ascuns cu 20 de neuroni, și un strat de ieșire cu 3 neuroni.

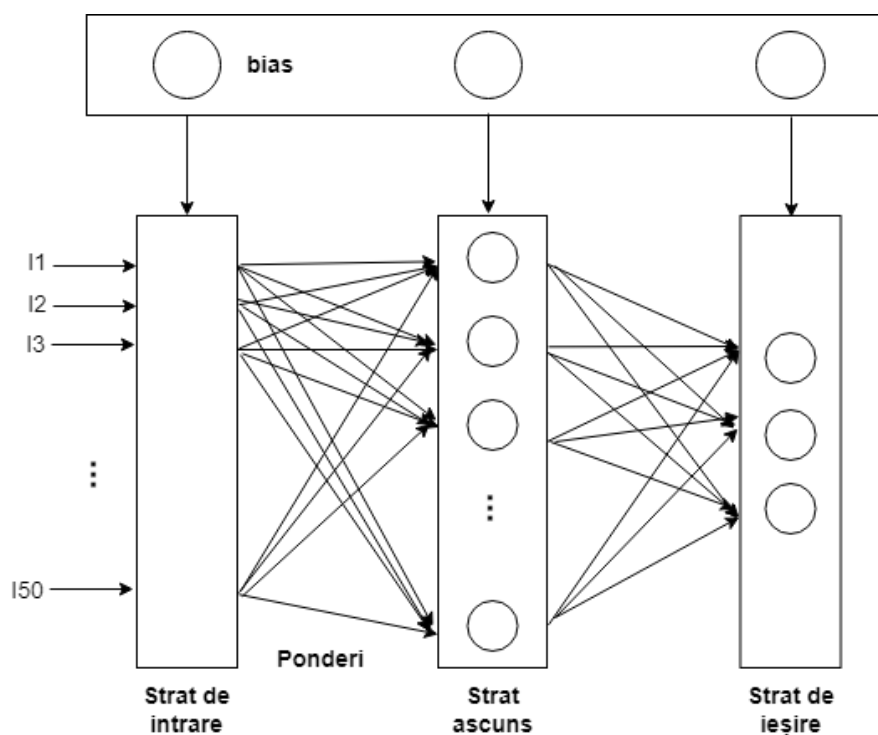


Figura 4.4: Modelul utilizat pentru detecția gesturilor

Datele de intrare sunt reprezentate de trei fișiere .csv care conțin diferite măsurări ale accelerației, 150 de eșantioane pentru fiecare măsurare. Primul fișier conține secvențe repetate ale ridicării mâinii, reprezentând gestul de interes pentru acest sistem. Acesta a fost colectat de la un număr de 14 persoane, cu vârste cuprinse între 18 și 74 de ani (7 persoane cu vârste cuprinse între 18-24, 2 persoane cu vârste cuprinse între 24-44, 5 persoane cu vârstă cuprinsă între 44-74). Al doilea fișier conține secvențe cu mișcări oarecare în jurul corpului, iar ultimul fișier conține secvențe cu mâna aflată în stare de repaus, la diferite amplitudini.

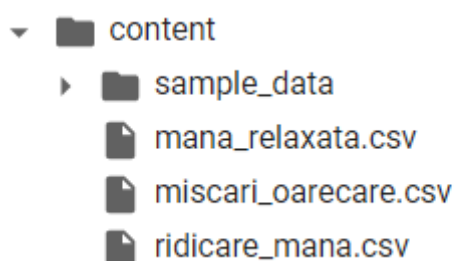


Figura 4.5: Fișierele cu datele de intrare

Primul pas în elaborarea algoritmului a fost crearea matricei pentru ieșiri. Aceasta este o matrice cu dimensiunea 3x3, având 1 pe diagonala principală. Numărul de gesturi vor fi trei: ridicare mână, mișcări oarecare în jurul corpului și mână în repaus.

Legătura dintre vectorul de intrare și cel de ieșire (inputs și outputs) se poate observa în figura 4.7:

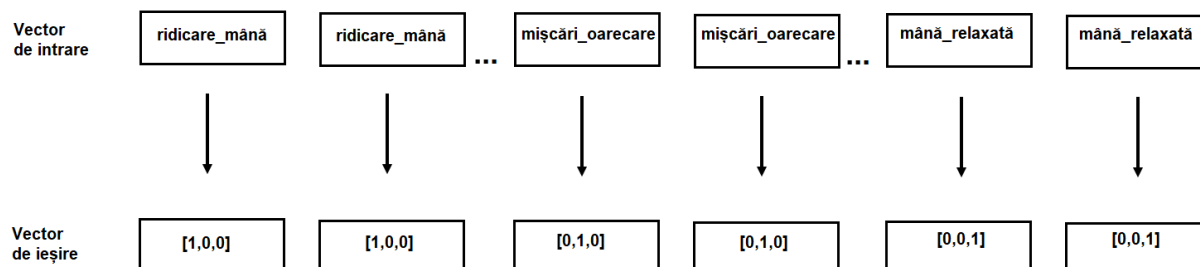


Figura 4.6: Legătura dintre elementele vectorului de intrare și elementele vectorului de ieșire

După acest pas, am amestecat datele folosind funcția `shuffle()`, iar apoi le-am împărțit în trei seturi: antrenare (60%), testare (20%) și validare (20%). Ulterior acestui pas, am creat și antrenat modelul.

### Construirea modelului:

- Stratul Dense este utilizat pentru ca fiecare neuron dintr-un strat să fie conectat cu toți neuronii din stratul următor;
- Funcția de activare ReLU (Rectified Linear Unit) – `activation = 'relu'` pentru stratul de intrare și stratul ascuns: returnează 0 dacă primește orice intrare negativă, și valoarea respectivă pentru intrare pozitivă;
- Funcția de activare softmax (`activation='softmax'`) pentru stratul de ieșire: aceasta generează un vector ale cărui elemente sunt distribuții de probabilitate, suma lor fiind egală cu 1. Deoarece sistemul trebuie să clasifice rezultatul în unul din cele trei gesturi, această funcție de activare o să returneze procentul în care fiecare gest se regăsește în rezultat.

### Compilarea modelului:

- Algoritmul de optimizare Adam actualiză ponderile rețelei iterativ pe baza datelor de antrenare.
- Pentru evaluare am folosit `'binary _crossentropy'`. Aceasta se utilizează pentru estimarea erorii modelului
- Pentru testarea acurateții am folosit `'accuracy'`

### Antrenarea modelului:

- O epocă reprezintă un ciclu complet de antrenament pe setul de antrenare. Numărul optim de epoci este atunci când eroarea converge sau scade sub un anumit prag;

- `Batch_size`: trebuie să fie mai mare sau egal cu 1 și mai mic sau egal cu numărul de eșantioane din setul de date de antrenare;

### **Convertirea modelului antrenat în Tensorflow Lite:**

- Cu ajutorul funcției `'convert()'`, un model antrenat va fi convertit într-un fișier de tipul `.tflite`. Acest fișier va fi folosit pentru generarea fișierului header care va fi utilizat pentru programarea dispozitivului în Arduino IDE.

### **Generarea fișierului header:**

- Se generează un fișier header `model.h` care creează o matrice constantă de octeți ce conține modelul `tflite`.

#### **4.1.3 Programarea sistemului de inferență al rețelei neuronale**

Programarea sistemului de inferență al rețelei neuronale pentru detectarea gestului de ridicare a mâinii la gură în procesul de ingestie a medicamentului are patru părți principale:

- Importarea modelului rețelei neuronale (fișierul header) și a bibliotecii Tensorflow Lite pentru ESP32
- Preluarea datelor de la accelerometru
- Detecția gesturilor pe baza datelor preluate de la accelerometru și a modelului importat
- Conexiunea Bluetooth cu aplicația mobilă pentru a-i putea transmite dacă gestul a fost detectat în intervalul destinat administrării tratamentului

## **4.2 Dozator inteligent de medicamente EZ Pills**

EZ Pills este un dozator automat de medicamente care ajută utilizatorii să urmeze corect tratamentul prescris de medicul lor deoarece: notifică cu o alarmă la orele la care trebuie să se administreze tratamentul medicamentos, eliberează automat medicamentul care trebuie administrat și interacționează cu o aplicație mobilă și cu un sistem de detecție al gesturilor sub formă de brățară care urmăresc complianța la tratamentul medicamentos ambulatoriu.

Acesta este un dispozitiv cu dimensiuni relativ mari (38cmx28cmx33cm), nefiind portabil.





Figura 4.7 : Dozatorul de medicamente EZ Pills

#### 4.2.1 Resurse hardware

##### Raspberry Pi 3 Model B+

Pentru a implementa dozatorul inteligent de medicamente, am ales să folosesc Raspberry Pi 3 Model B+, deoarece este un computer mic și accesibil, care are preț redus și performanță ridicată, astfel putând îndeplini toate cerințele necesare realizării proiectului.

Raspberry Pi este un dispozitiv de tip single-board computer care are ca sistem de operare Raspberry Pi OS, bazat pe Linux. Lansat în 2012, de-a lungul timpului au existat mai multe generații de Raspberry Pi: Pi de la 1 la 4 (majoritatea având un model A și un model B), și chiar un Pi 400 (lansat în 2021).

Dispozitivul utilizat în acest proiect este Raspberry Pi 3 Model B+. Acesta dispune de un procesor Broadcom BCM2837B0, cu o arhitectură ARM pe 64 de biți, ce rulează la o frecvență de 1.4GHz, într-o arhitectură de tip quad-core (4 nuclee). De asemenea, acesta pune la dispoziție memorie RAM de 1 GB și conectivitate: wireless dual-band (2.4 GHz și 5 GHz), Bluetooth 4.2 și BLE și Ethernet, putând transfera date la o viteză de până la 300 Mbps [22].

Din punct de vedere structural, acesta are un header GPIO cu 40 de pini, 4 porturi USB, conectori HDMI și MIPI CSI (Display Serial Interface și Camera Serial Interface) și slot pentru card microSD pentru rularea sistemului de operare Raspberry Pi OS și stocarea datelor.





*Figura 4.8: Raspberry Pi 3 Model B+ [22]*

### **Ecran tactil**

Ca interfață cu utilizatorul pentru dozatorul de medicamente, am optat pentru un ecran tactil LCD de 3.5", cu rezoluție de 480x320 pixeli, proiectat pentru Raspberry Pi. Interfața este una simplă și prietenoasă, datorită ecranului tactil, făcând-o potrivită și pentru persoanele vârstnice, principalii utilizatori ai dispozitivului.

Protocolul de comunicare este SPI, iar touchscreen-ul este de tip rezistiv. După instalarea driver-ului corespunzător acestuia, a necesitat calibrare, folosind un program numit xinput\_calibrator.

### **Servomotoare**

Pentru eliberarea medicamentelor din recipiente, am folosit 4 servomotoare de tip SG90 de 180°. Un factor esențial în alegerea acestui tip de servomotoare a fost dimensiunea acestora, 22.2 x 11.8 x 31 mm, deoarece era necesar ca acestea să intre în interiorul mecanismului proiectat. De asemenea, dispozitivul trebuie să fie unul compact, iar utilizarea unor componente de dimensiuni mari ar duce la creșterea dimensiunii dozatorului.

Servomotorul este proiectat pentru aplicațiile de mică putere. Acesta nu e supus unor sarcini mari, prin urmare nu e nevoie de un cuplu mare. Avantajul adus de această caracteristică este consumul redus de curent.

Pentru controlul servomotoarelor folosind un număr minim de pini (2 pini: SDA și SCL) am ales să folosesc driver-ul PCA9685. Acesta este un controller PWM cu 16 canale, având o rezoluție de 12 biți, iar comunicarea se face printr-o interfață I2C .

### **Cameră Raspberry Pi**

Pentru a se face verificarea că pacientul și-a luat medicamentul, s-a folosit o cameră Raspberry Pi care trimite fotografia bazei de date după ce s-a detectat mișcarea de ducere

a mâinii la gură, astfel încât supraveghetorul să poate vedea dacă pastila a fost luată sau nu.

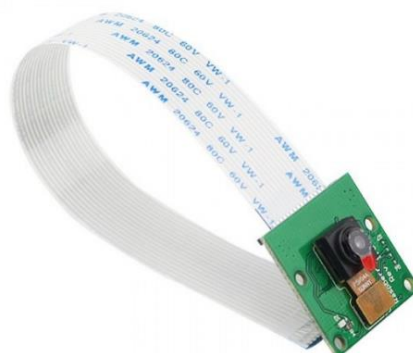


Figura 4.9 : Camera Raspberry Pi v1.3 [23]

### Senzori IR

Pentru a vedea dacă pastila a trecut prin orificiul prevăzut pentru eliberarea medicamentului, am utilizat modulul senzor infraroșu pentru detectarea obstacolelor, unul pentru fiecare recipient de medicamente. Senzorul se bazează pe reflexia radiației IR. Această radiație este emisă de către un led și este recepționată de către o fotodiodă.

Output-ul senzorului este digital (0 sau 1), iar distanța (2cm - 30cm) este reglată de potențiometrul.

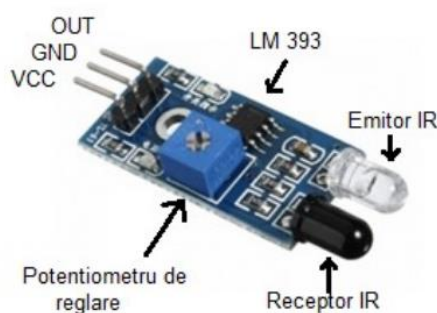


Figura 4.10: Modul senzor infraroșu pentru detectarea obstacolelor [24]

### Difuzor și amplificator

Pentru notificarea sonoră sub formă de alarmă, am avut nevoie de un difuzor și de un amplificator.

### Leduri

Pentru procesul de reumplere a recipientelor cu medicamente, am folosit 4 leduri. Când un medicament este selectat pentru adăugarea pastilelor, ledul corespunzător recipientului se va aprinde.

## 4.2.2 Conectarea componentelor

Pentru implementarea dozatorului inteligent de medicamente, componenta principală este dispozitivul Raspberry Pi 3 Model B+. La acest dispozitiv sunt conectate mai multe componente: un ecran tactil, un driver pentru controlul servomotoarelor, 4 senzori cu infraroșu pentru detectarea obstacolelor, o cameră Raspberry P, difuzorul și amplificatorul, dar și 4 leduri.

Pinii dispozitivului Raspberry Pi sunt:

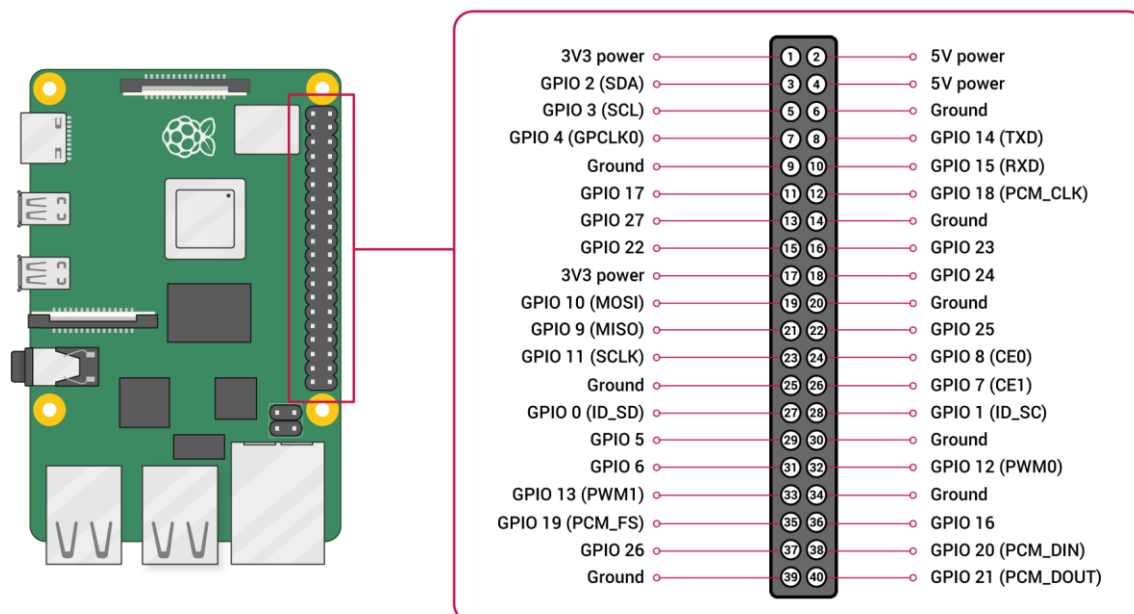


Figura 4.11: Pinii dispozitivului Raspberry Pi [25]

Ecranul tactil utilizează 17 pini ai dispozitivului Raspberry Pi: pinii 1,17 (alimentare 3.3V), pinii 2,4 (alimentare 5V), pinii 6, 9, 14, 20, 25 (GROUND), pinii 11, 18, 19, 21, 22, 23, 24, 26.

Driverul PCA9685 pentru controlul servomotoarelor se conectează la pinii 3 (pin SDA -> pin 3) și 5 (pin SCL -> pin 5) ai dispozitivului Raspberry Pi. PCA9685 comunică cu Raspberry Pi prin intermediul protocolului I2C. Pinul SDA al driverului are rol în transmiterea datelor seriale, iar pinul SCL în sincronizarea datele transmise. Pentru alimentare, se utilizează pinul 1 (3.3V), iar pentru conectarea la masă se utilizează pinul 6.

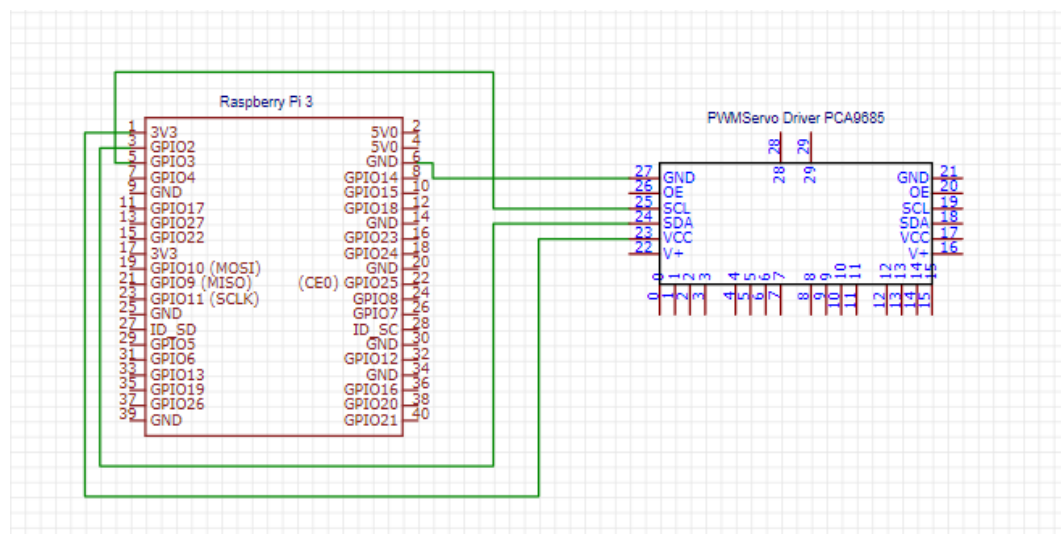


Figura 4.12 : Schema de conectare a driverului PCA9685 la Raspberry Pi

Servomotoarele sunt conectate la driverul PCA9685. Fiecare servomotor are 3 pini: pinul de generare a semnalului PWM, pinul VCC și pinul GROUND, conectați conform Figurii 4.13 . Servomotoarele sunt alimentate separat la o tensiune de 5V.

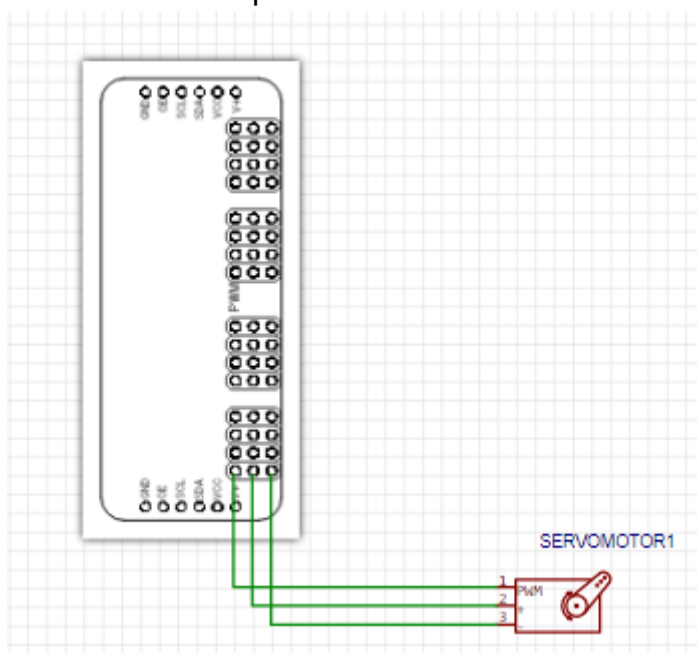


Figura 4.13 : Schema de conectare a unui servomotor la driverul PCA9685

Senzorii cu infraroșu pentru detectarea obstacolelor sunt conectați la dispozitivul Raspberry Pi astfel: pin OUT senzor 1 – GPIO 21, pin OUT senzor 2 – GPIO 20, pin OUT senzor 3 – GPIO 16, pin OUT senzor 4 – GPIO 26. Pinii GROUND sunt conectați la pinul 16 al dispozitivului Raspberry Pi, iar pinii VCC la pinul 1.

Camera Raspberry Pi este conectată la conectorul CSI al dispozitivului.

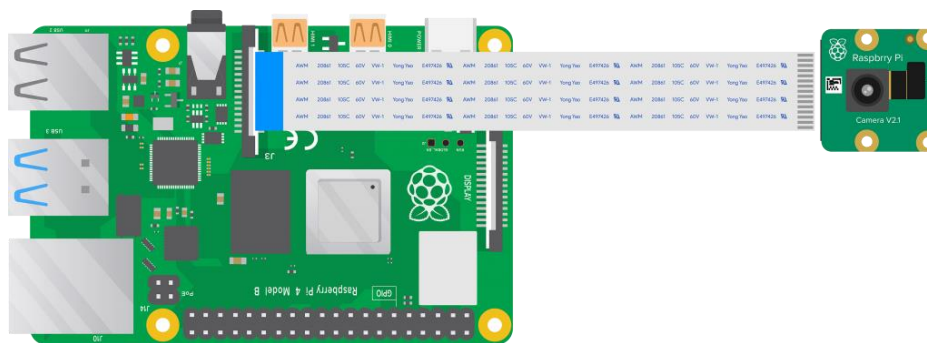


Figura 4.14 : Schema de conectare a camerei la Raspberry Pi

Difuzorul este conectat la portul jack al dispozitivului Raspberry Pi. Amplificatorul este alimentat separat la o tensiune de 5V.

#### 4.2.3 Software *embedded*

Pentru programarea dispozitivului Raspberry Pi am folosit ca limbaj de programare Python. În cadrul proiectului, am utilizat diferite biblioteci, precum:

- tkinter – biblioteca standard pentru crearea interfețelor grafice în Python
- pyrebase – pentru manipularea datelor din baza de date Firebase
- Adafruit ServoKit – pentru controlul servomotoarelor
- RPi.GPIO – pentru controlul pinilor GPIO ai dispozitivului Raspberry Pi
- picamera – pentru capturarea fotografiilor utilizând camera Raspberry Pi

#### Stocarea și sincronizarea datelor

Pentru stocarea și sincronizarea datelor precum: ora următoarei alarme, medicamentele asiguate pacientului, numărul de medicamente rămase în recipiente și fotografiile capturate de camera Raspberry Pi am utilizat Firebase Realtime Database și Firebase Storage. Motivul principal pentru care am ales această bază de date este că aplicația mobilă, prin intermediul căreia se setează medicația pacientului, salvează datele în Firebase.

Conexiunea cu baza de dată Firebase este realizată astfel:

```
config = {
    "apiKey": "uWBY4vknbyWfNBEFedjufDn4iNAA7ehf70pOdJZ8",
    "authDomain": "medtracking-8b822.firebaseio.com",
    "databaseURL": "https://medtracking-8b822-default-
rtddb.firebaseio.com",
    "storageBucket": "medtracking-8b822.appspot.com"
}

firebase = pyrebase.initialize_app(config)
```

```
db = firebase.database()
storage = firebase.storage()
```

Baza de date conține informații despre medicația tuturor utilizatorilor aplicației mobile. Această aplicație are rolul de a seta medicația utilizatorilor și de a îi notifica la orele la care trebuie să administreze tratamentul medicamentos. Aplicația poate funcționa și fără cutii inteligente de medicamente, în condițiile în care utilizatorii nu au nevoie de dozarea automată a pastilelor. Legătura dintre aplicație și cutia inteligentă de medicamente se face prin introducerea în aplicație a unui cod unic al cutiei, cod ce este afișat pe ecranul dozatorului. Odată realizată conexiunea dintre dispozitive, se va crea în baza de date o secțiune specială pentru fiecare cutie care conține informațiile următoare: dacă medicamentul a fost eliberat din cutie după ce a sosit ora administrării tratamentului, dacă a apărut o eroare de funcționare la cutie când s-a încercat administrarea tratamentului, dacă medicamentul de la ora respectivă a fost luat, numele medicamentului ce se va administra la următoarea notificare pentru luarea tratamentului, ora la care se va administra următorul tratament, numele pozei efectuate ca dovadă că medicamentul a fost administrat și codul unic de indentificare al utilizatorului cutiei.

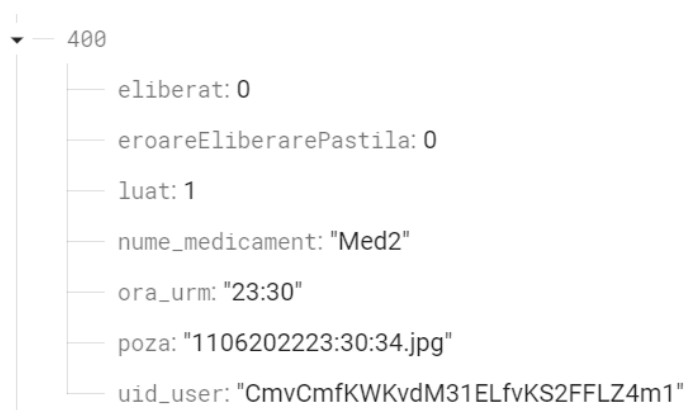


Figura 4.15 : Structura bazei de date asociată cutiei de medicamente

Secvența de cod pentru preluarea din baza de date a codul unic de identificare al utilizatorului dozatorului de medicamente:

```
uid=db.child(cod).child("uid").get().val()
```

Fiecare pacient are în baza de date secțiunea de medicament unde sunt stocate informațiile despre medicația sa (numele tuturor medicamentelor asiginate acelui pacient). Fiecare medicament are următoarele câmpuri: compartimentul în care se află în dozator și numărul de pastile rămase în acel compartiment (dacă este realizată conexiunea aplicație-dozator de medicamente), culoarea pilulei, data la care a fost adăugat medicamentul în planul de tratament, data ultimei administrări a medicamentului, intervalul dintre două doze și ora următoarei notificări pentru administrarea tratamentului.

Secvența de cod pentru preluarea din baza de date a medicamentelor asignate utilizatorului cutiei:

```
numeMedicament=db.child("medicament").child(uid).get()  
for i in numeMedicament.each():  
    medicament.append(i.key())
```

Fotografiile capturate de camera Raspberry Pi vor fi salvate în Firebase Storage, într-un folder care are ca nume codul unic de indentificare al pacientului. Numele fotografiei va fi reprezentat de contopirea datei cu ora, minutul si secunda la care se efectuează fotografia.

```
#salvarea fotografiilor capturate de cameră în Firebase Storage  
moment_actual=datetime.now().strftime("%d%m%Y%H%M%S")  
nume_image=moment_actual+".jpg"  
camera.capture(nume_image)  
storage.child(uid).child(nume_image).put(nume_image)
```

### Interfața cu utilizatorul

Pentru interacțiunea cu utilizatorul, am creat o interfață grafică intuitivă cu dublu rol: rol informativ, afișând ora la care se va administra următorul tratament, dar și rol în reumplerea celor 4 recipiente de medicamente.

Această interfață are trei ecrane. Ecranul principal conține codul unic al cutiei (codul prin care se face legătura între aplicația mobilă și dozatorul inteligent de medicamente), ora următoarei alarme și un buton, numit „Reumplere recipiente”, care duce spre pagina următoare. Cea de-a doua pagină conține numele medicamentelor asignate pacientului prin intermediul aplicației, preluate din baza de date. Dacă se dorește reumplerea stocului unui medicament, se va apăsa pe numele acesuia pentru a fi redirecționat în cea de-a treia pagină unde se specifică câte medicamente se vor adăuga în recipientul respectiv. De asemenea, în cea de-a treia pagină este afișat și numărul medicamentelor rămase în recipient. Odată finalizată acțiunea utilizatorul dozatorului va trebui să adauge pastilele în recipient.

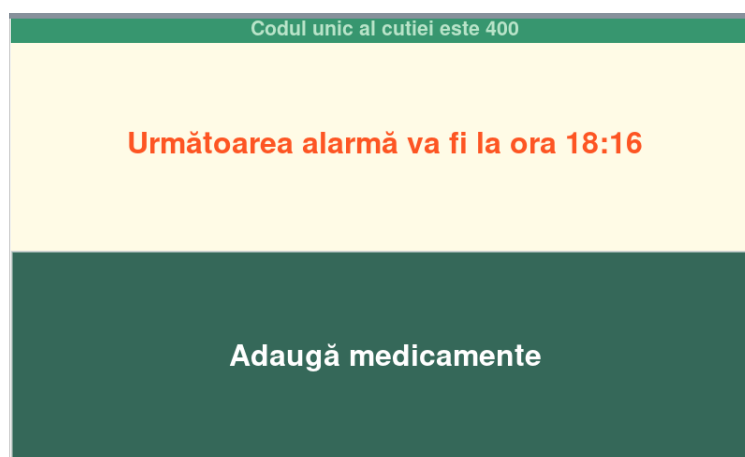


Figura 4.16 : Ecranul principal al dozatorului de medicamente



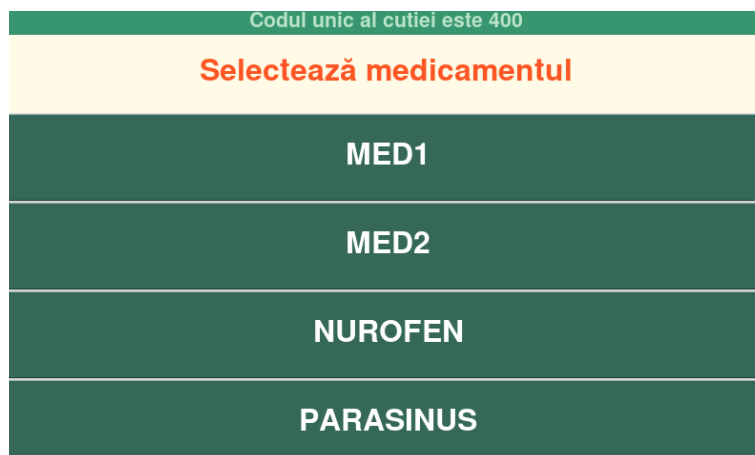


Figura 4.17 : Cel de-al doilea ecran al dozatorului de medicamente

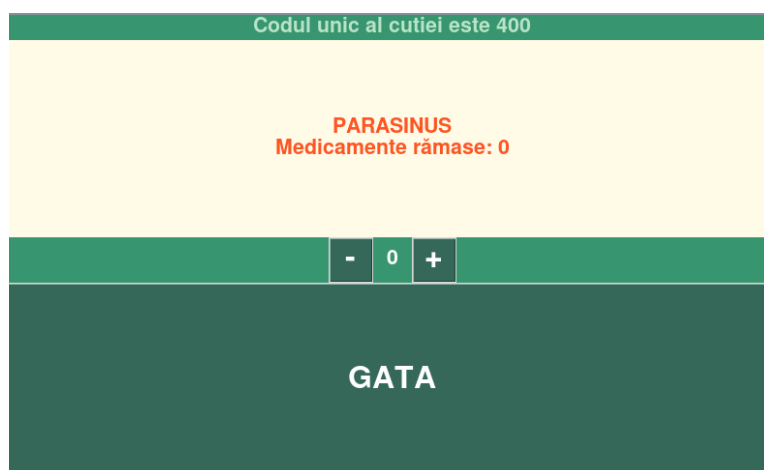


Figura 4.18: Cel de-al treilea ecran al dozatorului de medicamente

### Programarea alarmelor și eliberarea pastilelor

Pe lângă dozarea automată a pastilelor, dozatorul inteligent de medicamente dispune de o alarmă care va începe în momentul în care trebuie administrat tratamentul medicamentos și se încheie fie dacă tratamentul a fost administrat, fie dacă a trecut intervalul alocat administrării acestuia.

După ce alarma a fost inițiată, se va prelua din baza de date medicamentul care trebuie eliberat. Servomotorul care controlează rotirea capacului de la recipientul cu medicamente în vederea eliberării unei pastile începe să funcționeze, iar din acel moment modulul cu senzori infraroșu pentru detectarea obstacolelor verifică în continuu dacă pastila a trecut prin orificiul prevăzut pentru eliberarea pastilei. În cazul în care există medicamente în recipient, dar o pastilă nu a fost eliberată după prima încercare, se va mai încerca procesul de eliberare. Procesul de eliberare a medicamentelor din dozator este ilustrat în Figura 4.19.



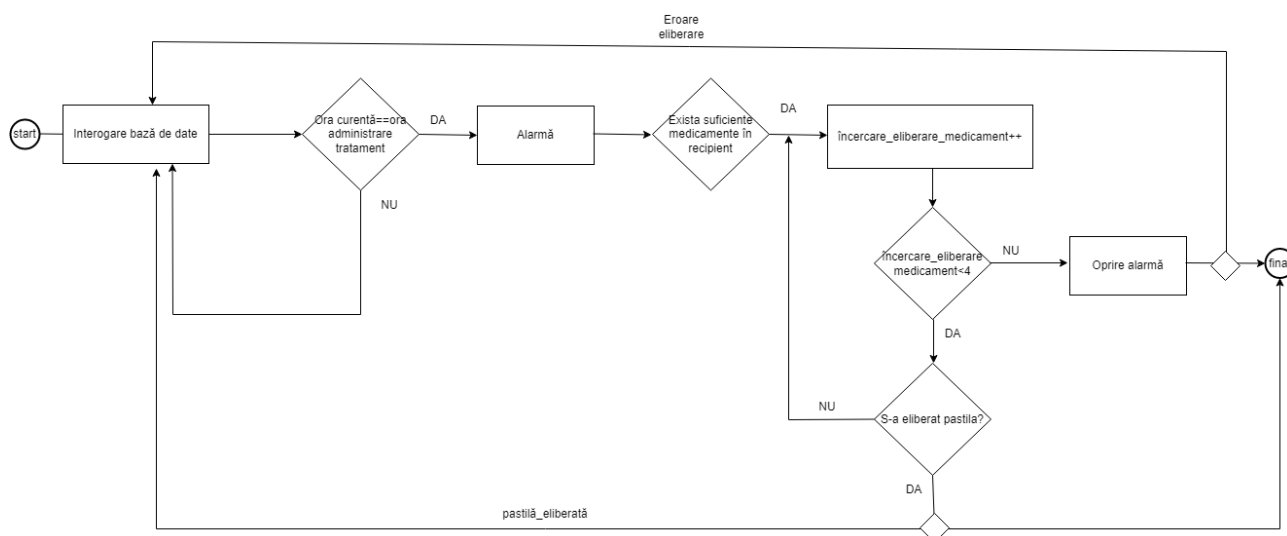


Figura 4.19 : Procesul de eliberare a medicamentelor din dozatorul inteligent

### Oprirea alarmelor

Alarmerle, odată inițiate, se pot opri în două moduri: fie utilizatorul și-a luat medicamentul (pastila a fost eliberată, iar în timpul acordat administrării tratamentului s-a detectat gestul de ducere a mâinii la gură) sau a trecut intervalul destinate luării medicamentului (acest interval este de 5 minute de la începerea alarmei). Există o abatere de la aceste reguli pentru oprirea alarmei, iar aici se face referire la funcționarea corespunzătoare a dozatorului. În cazul în care dozatorul nu funcționează corect, adică în urma celor trei încercări nu s-a putut realiza eliberarea pastilei din dozator, alarma se va opri automat, marcându-se o eroare de funcționare (marcare eroare eliberare pastilă în baza de date). Aplicația va anunța supraveghetorul utilizatorului (dacă există), iar problema va fi rezolvată de acesta.

Secvența de cod care realizează oprirea alarmelor este următoarea:

```

if medicamentAdministrat==1 or (datetime.strptime(current_time,
'%H:%M') - time_urmAlarma) == timedelta(minutes=5):
    #opresc alarma
    stopAlarm()
  
```

## 5. TESTAREA SISTEMULUI

### 5.1 Testarea sistemului de detecție a gesturilor

Pentru a testa acuratețea sistemului de detecție a gesturilor, am efectuat mai multe teste.

Primul test este efectuat pe seturile de date de antrenare și de validare. Modelul pare să învețe bine, având o precizie de peste 99%, atât pentru datele de antrenare, cât și pentru cele de validare. Funcția de măsurare a erorii utilizată este *binary\_crossentropy*. Eroarea modelului (pierderile) este mai mică decât 0.05. Reprezentarea grafică a datelor s-a realizat cu ajutorul bibliotecii *matplotlib*.

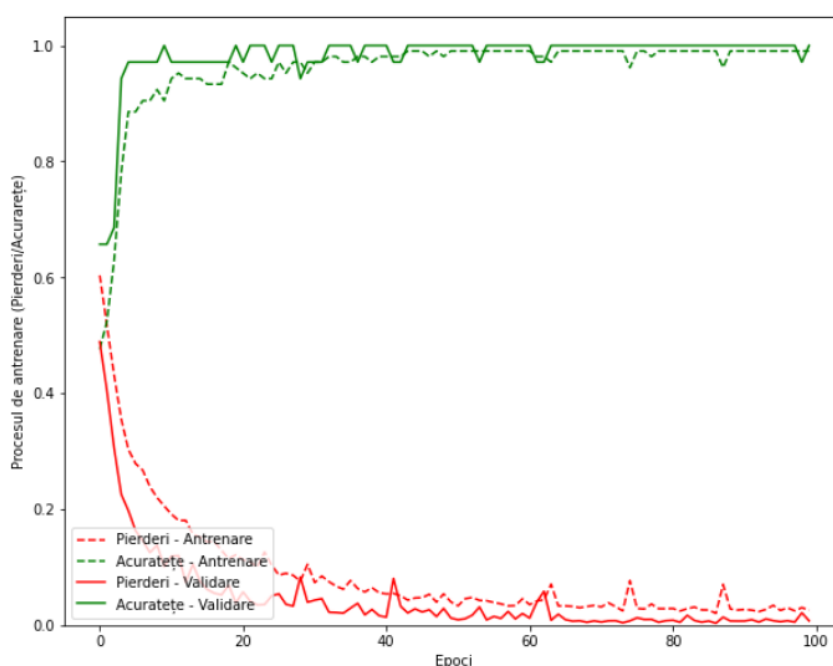


Figura 5.1: Reprezentarea grafică a erorii pentru funcția de evaluare și a acurateții pentru funcția de acuratețe, atât pentru datele de antrenare, cât și pentru datele de validare

Al doilea test a fost efectuat pe setul de date de testare. Acuratețea modelului este 100%, iar pierderile sunt de aproximativ 0.002.

```
▶ loss, accuracy = model.evaluate(inputs_test, outputs_test, batch_size = 10, verbose = 1)
print("Loss - test :", loss)
print("Accuracy - test :", accuracy)
```

4/4 [=====] - 0s 6ms/step - loss: 0.0025 - accuracy: 1.0000  
Loss - test : 0.002463326323777437  
Accuracy - test : 1.0

Figura 5.2: Testarea pe setul de date de testare

Al treilea test a fost efectuat practic, pe 5 persoane, două dintre ele fiind cele de la care am colectat o parte din date, iar celelalte 3 fiind persoane noi pentru a putea testa cât mai bine precizia de funcționare a algoritmului. Fiecare persoană a efectuat 5 teste, iar fiecare test a presupus 10 ridicări de mână asociate ingestiei unei pastile. Rezultatul testului a fost media aritmetică a procentelor de recunoaștere a gestului de ridicare a mâinii din cele 10 încercări. La fel ca la colectarea datelor, s-au folosit produse tip bomboane pentru a substitui medicamentele.

Tabelul 5.1: Testarea practică a acurateții dispozitivului de detecție a gesturilor prin efectuarea a 5 teste/persoană, 10 încercări/test la un număr de 5 persoane, M fiind media aritmetică a celor 10 încercări

	Test 1	Test 2	Test 3	Test 4	Test 5
Persoana 1	0.989802	1.000000	0.999823	0.996998	0.999983
	1.000000	0.999974	0.999999	0.998863	0.857413
	0.996036	1.000000	1.000000	0.999967	1.000000
	0.990195	0.866578	0.999650	0.999969	0.960832
	1.000000	0.992302	1.000000	1.000000	0.990579
	0.999648	0.999966	0.998800	0.999511	1.000000
	0.999991	0.999999	0.999395	1.000000	0.999999
	1.000000	0.934785	0.987594	0.999926	0.978256
	0.999964	1.000000	1.000000	0.999873	0.999996
	0.999671	0.972247	0.999999	0.991738	1.000000
	M=0.997522	M=0.9765851	M=0.998526	M=0.998684	M=0.978795
Persoana 2	1.000000	1.000000	0.989976	1.000000	0.999886
	0.999998	0.999648	0.998894	0.999976	0.998652
	0.970230	1.000000	1.000000	0.991948	0.999549
	1.000000	0.999930	1.000000	1.000000	1.000000
	1.000000	0.999999	0.998131	0.999511	1.000000
	0.998972	0.999996	0.978722	0.972337	1.000000
	0.979291	0.999988	0.999998	0.996116	0.988832
	0.999893	1.000000	0.999966	1.000000	0.990021
	1.000000	1.000000	0.992322	0.989855	0.999853
	1.000000	0.987996	0.999999	0.990579	0.999648
	M=0.994838	M=0.998755	M=0.995800	M=0.994032	M=0.997644
Persoana 3	0.999996	0.969864	1.000000	0.999988	0.988948
	0.987992	0.988995	0.979868	0.992209	0.999986
	0.989980	0.990224	0.979980	0.989943	0.981342
	1.000000	0.998893	0.988656	0.992745	0.965998
	0.999998	1.000000	0.992921	0.999999	0.989766
	0.999968	0.998987	1.000000	0.998972	0.990906

	0.999297	0.979304	0.990924	0.999291	0.984011
	0.999895	0.993209	0.988993	0.999453	0.999294
	1.000000	0.999893	0.986889	0.999996	0.998469
	0.999893	0.998887	0.989973	0.999988	0.998893
	M=0.997701	M=0.991821	M=0.989814	M=0.997258	M=0.989826
Persoana 4	0.999886	0.997546	0.988775	0.999648	0.992302
	0.999998	0.896578	0.888998	0.999991	0.999966
	0.986789	0.989882	0.989999	0.986789	0.998131
	0.969782	0.989905	0.999965	0.969782	0.978722
	0.990029	0.999996	0.990025	0.999754	0.891429
	0.988741	0.989976	0.990189	0.896578	0.988741
	0.999384	0.999882	0.999648	1.000000	0.999384
	0.999887	0.998801	0.999996	0.960832	0.999930
	0.999999	0.999999	0.999999	0.999650	0.999999
	0.989964	0.999999	1.000000	1.000000	0.989912
	M=0.994925	M=0.986256	M=0.984759	M=0.981302	M=0.983961
Persoana 5	0.989802	0.999648	0.990029	1.000000	0.988948
	1.000000	0.999991	0.999998	0.999991	0.999986
	0.866578	0.998972	0.970230	0.999754	0.999921
	0.992302	0.979291	0.989882	0.990924	0.977233
	0.999395	0.891998	1.000000	0.979304	0.992302
	0.987594	0.989999	0.998972	0.999966	0.998972
	0.979291	0.986789	0.990025	0.979341	0.988741
	0.999893	0.969782	0.984011	0.999903	0.999993
	0.978256	0.993209	0.999968	0.988993	0.998131
	0.999996	0.999893	0.969782	0.984023	0.999029
	M=0.979310	M=0.980957	M=0.989289	M=0.992213	M=0.994325

Conform datelor din tabel, se poate observa că algoritmul funcționează corespunzător, gestul de ridicare a mâinii fiind recunoscut cu o precizie foarte mare.

## 5.2 Testarea dozatorului inteligent de medicamente

Primul test efectuat dozatorului de medicamente a fost pentru a vedea câte eliberări de pastile vor avea succes din 100 de încercări de eliberare.

Inițial, structura dozatorului era diferită, iar în încercarea de eliberare a medicamentelor, din cauza faptului că acestea rămâneau blocate în orificiul prevăzut pentru eliberare sau nu erau preluate atunci când servomotoarele încercau eliberarea, doar 80/100 de medicamente erau eliberate din dozator.

Ulterior, după modificarea structurii, din 100 de încercări de eliberare a medicamentului, 95 au fost cu succes. În acele încercări eșuate, pastilele au rămas în pâlnia prevăzută în interiorul dozatorului, după ce au ieșit din compartimente.

Un alt test efectuat dozatorului a fost pentru a vedea dacă într-o încercare de eliberare a medicamentului sunt eliberate mai multe medicamente decât ar trebui. Deoarece se fac trei încercări de eliberare în cazul în care medicamentele nu sunt preluate din prima de componenta controlată de servomotor, pentru a detecta dacă pastilele au trecut prin orificiul prevăzut pentru eliberare au fost utilizați senzorii IR pentru detectarea obstacolelor. Inițial, nefiind calibrați pentru scopul prevăzut în proiect (un senzor este emitor IR, iar celălalt receptor IR, iar când trece pastila printre cei doi senzori, unda este întreruptă, considerându-se că medicamentul a fost eliberat), nu detectau trecerea pastilei corespunzător. După calibrare, pastilele sunt detectate corect, 99 din 100 de încercări având succes.

### **5.3 Testarea întregului sistem**

Funcționalitățile întregului sistem au fost testate prin efectuarea practică a 30 de teste. S-au verificat: sincronizarea dispozitivelor, setarea alarmelor (atât alarma dozatorului, cât și alarma aplicației mobile cu care sistemul interacționează), conexiunea Bluetooth dintre aplicație mobilă și sistemul de detecție al gesturilor și detecția gestului de ingestie a medicamentului. În urma celor 30 de teste reușite, s-a confirmat că sistemul funcționează corespunzător.

## 6. CONCLUZII

### 6.1 Realizări

Pentru realizarea lucrării, multiple surse ale literaturii de specialitate au fost consultate pentru a avea o privire de ansamblu mai largă și a analiza mai multe variante de implementare. Soluția finală a fost aleasă în concordanță cu echipamentele accesibile, timpul, obiectivele și costurile implicate.

Etapele parcurse pentru elaborarea lucrării:

- Investigarea problemei existente cu care se confruntă pacienții atunci când urmează tratamentul ambulatoriu și găsirea unei soluții pentru rezolvarea ei
- Analiza variantelor de proiectare și implementare, dar și componentele necesare realizării practice
- Colectarea seturilor de date pentru a putea antrena și testa algoritmul de Machine Learning
- Crearea algoritmului de Machine Learning pentru detecția gesturilor
- Dezvoltarea sistemului care cuprinde un dozator inteligent de medicamente și un sistem de detecție a gesturilor
- Testarea sistemului

Pentru realizarea practică s-a ținut cont de raportul cost-performanță, dar a fost luată în calcul și alegerea unei componente cu consum redus de energie pentru dispozitivul atașat încheieturii mâinii pentru a fi potrivit utilizării zilnice.

Pentru antrenarea algoritmului de Machine Learning s-au colectat seturi de date de la 14 persoane cu vârste diferite, iar pentru testarea acestuia datele au fost colectate de la 5 persoane, două dintre ele fiind aceleași de la care am colectat o parte din datele pentru antrenare. În procesul de colectare a datelor s-au utilizat produse de tip bomboane pentru a putea simula procesul de ingestie a medicamentelor.

Făcând o comparație cu obiectivele descrise în subcapitolul 1.2 cu ceea ce s-a realizat în această lucrare, se pot concluziona următoarele:

- Am identificat care sunt cerințele funcționale ale unui sistem menit să vină în ajutorul persoanelor care urmează un tratament medicamentos ambulatoriu. Acest sistem este ideal chiar și persoanelor cu boli ale sistemului nervos care se manifestă prin pierderi de memorie;
- Am implementat un sistem care îndeplinește toate cerințele funcționale specificate: setarea alarmelor și dozarea automata a pastilelor la orele destinate administrării tratamentului medicamentos, detecția ingestiei pastilei

prin intermediul unui dispozitiv de detecție a gesturilor care are la bază un algoritm de Machine Learning, capturarea fotografiei cu recipientul unde au fost eliberate medicamentele după ce procesul de ingestie a fost detectat pentru verificarea suplimentară că medicamentele au fost luate, dar și interconectarea tuturor componentelor sistemului (dozator inteligent, sistem de detecție a gesturilor, aplicație mobilă Android cu rol în setarea medicației);

- Am realizat antrenarea algoritmului de Machine Learning folosind seturi de date colectate de la 14 persoane;
- Am realizat testarea algoritmului de Machine Learning
- Soluția abordată este potrivită și pentru monitorizarea de la distanță a persoanelor care urmează un tratament ambulatoriu de către persoanele care se ocupă cu îngrijirea și supravegherea acelor pacienți.

## 6.2 Dezvoltări ulterioare

Chiar dacă s-au realizat toate obiectivele propuse, sistemului i se pot aduce îmbunătățiri:

- Extinderea numărului de compartimente ale dozatorului de medicamente
- Implementarea unui algoritm de Machine Learning care să detecteze din fotografiile realizate de cameră dacă există medicamente în recipient după ce ingestia medicamentelor a fost detectată
- Setarea medicației prin intermediul interfeței grafice a dozatorului

## BIBLIOGRAFIE

- 1] | „România - Profilul de țară din 2021 în ceea ce privește sănătatea,” State of Health in the EU, [Interactiv]. Available: [https://ec.europa.eu/health/system/files/2022-01/2021\\_chp\\_romania\\_romanian.pdf](https://ec.europa.eu/health/system/files/2022-01/2021_chp_romania_romanian.pdf). [Accesat 15 Mai 2022].
- 2] | G. 2. D. F. Collaborators, „Estimation of the global prevalence of dementia in 2019 and forecasted prevalence in 2050: an analysis for the Global Burden of Disease Study 2019,” *Lancet Public Health*, vol. 7, 2022.
- 3] | „The Cost of Not Taking Your Medicine,” The New York Times, [Interactiv]. Available: <https://www.nytimes.com/2017/04/17/well/the-cost-of-not-taking-your-medicine.html>. [Accesat 15 Mai 2022].
- 4] | I. Sommerville, „Software Engineering,” *England: Addison-Wesley*, vol. 9th ed. Harlow, 2010.
- 5] | „Automatic Pill Dispenser Evaluations: Hands-on,” 15 Mai 2022. [Interactiv]. Available: <https://www.techenhancedlife.com/citizen-research/automatic-pill-dispenser-evaluations>.
- 6] | „MedaCube Medication Dispenser: Review,” Tech-enhanced Life, [Interactiv]. Available: <https://www.techenhancedlife.com/reviews/medacube-medication-dispenser>. [Accesat 15 Mai 2022].
- 7] | „MedMinder, Maya: Review,” Tech-enhanced Life, [Interactiv]. Available: <https://www.techenhancedlife.com/reviews/medminder-maya>. [Accesat 15 Mai 2022].
- 8] | „Hero Medication Dispenser: Review,” Tech-enhanced Life, [Interactiv]. Available: <https://www.techenhancedlife.com/reviews/hero-medication-dispenser>. [Accesat 15 Mai 2022].
- 9] | „3 Best Automatic Pill Dispensers in 2022: Reviews Ad Ultimate Buying Guide,” Terrycralle, [Interactiv]. Available: <https://www.terrycralle.com/best-automatic-pill-dispensers/>. [Accesat 16 Mai 2022].
- 10] | „MedaCube™ Automatic Pill Dispenser,” 15 Mai 2022. [Interactiv]. Available: <https://www.medacube.com/collections/frontpage/products/medacube>.
- 11] | „Much More than a Pill Dispenser,” MedMinder, [Interactiv]. Available: <https://www.medminder.com/>. [Accesat 15 Mai 2022].
- 12] | „Goodbye pill box, Hello Hero,” HeroHealth, [Interactiv]. Available: [https://herohealth.com/?sscid=51k6\\_f2j1t&utm\\_source=shareasale&utm\\_medium=affiliate&utm\\_campaign=1469987\\_1294058](https://herohealth.com/?sscid=51k6_f2j1t&utm_source=shareasale&utm_medium=affiliate&utm_campaign=1469987_1294058). [Accesat 16 Mai 2022].
- 13] | J. C. S. L. Y. J. Minwoo Kim, „IMU Sensor-Based Hand Gesture Recognition for Human-Machine Interfaces,” *Sensors*, 2019.
- 14] | Y.-J. J. T.-M. T. a. W.-J. H. Yen-Cheng Chu, „Recognition of Hand Gesture Sequences by Accelerometers and Gyroscopes,” *Applied Sciences*, vol. 10, p. 6507, 2020.
- 15] | „LILYGO® T-WATCH-2020,” Lilygo, [Interactiv]. Available: [http://www.lilygo.cn/prod\\_view.aspx?TypeId=50053&Id=1380&FId=t3:50053:3](http://www.lilygo.cn/prod_view.aspx?TypeId=50053&Id=1380&FId=t3:50053:3). [Accesat 26 Mai 2022].



- 16] | „LilyGO's Upgraded TTGO T-Watch-2020 Ditches the Bulk, Puts Display, Sensors, and ESP32 On Your Wrist,” hackster.io, [Interactiv]. Available: <https://www.hackster.io/news/lilygo-s-upgraded-ttgo-t-watch-2020-ditches-the-bulk-puts-display-sensors-and-esp32-on-your-wrist-22cdc19fb9d3>. [Accesat 25 Mai 2022].
- 17] | „Machine Learning,” IBM, [Interactiv]. Available: [https://www.ibm.com/cloud/learn/machine-learning#toc-how-machin-NoVMSZI\\_](https://www.ibm.com/cloud/learn/machine-learning#toc-how-machin-NoVMSZI_). [Accesat 15 Mai 2022].
- 18] | „ESP32 WebSocket Server: Control Outputs (Arduino IDE),” RANDOM NERD TUTORIALS, [Interactiv]. Available: <https://randomnerdtutorials.com/esp32-websocket-server-arduino/>. [Accesat 15 Mai 2022].
- 19] | „Python has become the most popular language, according to TIOBE,” Linuxiac, [Interactiv]. Available: <https://linuxiac.com/python-the-most-popular-programming-language/>. [Accesat 06 Iunie 2022].
- 20] | „About Keras,” Keras, [Interactiv]. Available: <https://keras.io/about/>. [Accesat 28 Mai 2022].
- 21] | „Neural networks Bias and Weight,” Medium, [Interactiv]. Available: <https://medium.com/fintechexplained/neural-networks-bias-and-weights-10b53e6285da>. [Accesat 20 Mai 2022].
- 22] | „Raspberry Pi 3 Model B+,” Optimus Digital, [Interactiv]. Available: [https://www.optimusdigital.ro/ro/placi-raspberry-pi/5091-raspberry-pi-3-model-b-plus.html?search\\_query=Raspberry+Pi+3+Model+B+&results=89](https://www.optimusdigital.ro/ro/placi-raspberry-pi/5091-raspberry-pi-3-model-b-plus.html?search_query=Raspberry+Pi+3+Model+B+&results=89). [Accesat 26 Mai 2022].
- 23] | „CAMERĂ VIDEO RASPBERRY PI,” Cleste, [Interactiv]. Available: [https://cleste.ro/camera-video-raspberry-pi.html?utm\\_medium=GoogleAds&utm\\_campaign=&utm\\_source=&gclid=Cj0KCQjw4uaUBhC8ARIsANUuDjUr0ZIGb8iFVjRBFEM-BDfv2CG9Uj-crnBz\\_jgl7l4f-MmwASzVWlaAsg1EALw\\_wcB](https://cleste.ro/camera-video-raspberry-pi.html?utm_medium=GoogleAds&utm_campaign=&utm_source=&gclid=Cj0KCQjw4uaUBhC8ARIsANUuDjUr0ZIGb8iFVjRBFEM-BDfv2CG9Uj-crnBz_jgl7l4f-MmwASzVWlaAsg1EALw_wcB). [Accesat 3 Iunie 2022].
- 24] | „Modul Senzor Infrarosu de Obstacole,” Optimus Digital, [Interactiv]. Available: [https://www.optimusdigital.ro/ro/senzori-senzori-optici/4514-senzor-infrarosu-de-obstacole.html?search\\_query=Modul+Senzor+Infrarosu+de+Obstacole&results=6](https://www.optimusdigital.ro/ro/senzori-senzori-optici/4514-senzor-infrarosu-de-obstacole.html?search_query=Modul+Senzor+Infrarosu+de+Obstacole&results=6). [Accesat 30 Mai 2022].
- 25] | „Raspberry Pi Documentation,” Raspberry Pi, [Interactiv]. Available: <https://www.raspberrypi.com/documentation/computers/os.html>. [Accesat 15 Mai 2022].



## **Anexe – Declarație de autenticitate a lucrării de finalizare a studiilor**