

**UNIVERSITATEA POLITEHNICĂ TIMIȘOARA
FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
DEPARTAMENTUL DE INFORMATICĂ**

**MONITORIZAREA LA DISTANȚĂ A PARAMETRILOR VITALI PRIN IoT ȘI
APLICAȚII MOBILE**

**Candidat:
DINICA ROXANA-DANIELA**

**Coordonator:
Ș.l. Dr. Ing. MIHAELA CRIȘAN-VIDA**

**Timișoara
2024–2025**

Cuprins:

Capitolul 1. Introducere.....	4
Capitolul 2. Studiul bibliografic și noțiunile teoretice	5
2.1 Necesitatea tehnologiei IOT	5
2.2 Senzorii utilizați	7
2.3 Avantajele sistemelor de notificare la distanță	9
2.4 Sisteme similare existente pe piața.....	11
2.5 Probleme ale sistemelor existente și oportunități de dezvoltare	12
Capitolul 3. Prezentarea sistemului propus spre dezvoltare și componentele folosite	14
3.1 Prezentarea sistemului propus spre dezvoltare	14
3.2 Schema bloc a sistemului.....	15
3.3 Costul de implementare	17
3.4 Componentele folosite	18
3.4.1 Placa de dezvoltare	18
3.4.2 Senzorul de temperatură și umiditate	20
3.4.3 Senzorul de puls	21
3.4.4 Senzorul pentru temperatura corpului	23
3.4.5 Modulul bluetooth	24
3.4.6 Ecranul LCD	27
Capitolul 4. Dezvoltarea componentei hardware a sistemului	29

Capitolul 5. Dezvoltarea componentei software	40
5.1 Configurarea mediului de dezvoltare	40
5.2 Schema logică	41
5.3 Secvențele programului	43
5.4 Dezvoltarea aplicației Android	55
Capitolul 6. Testarea și rezultatele experimentale	61
Concluzii.....	66
Bibliografie	67
Anexe	69
Anexa 1 Codul sursă al sistemului	69

Capitolul 1. Introducere

Motivația din spatele acestei lucrări provine din nevoia constantă de a îmbunătăți monitorizarea stării de sănătate a oamenilor, într-un mod accesibil și eficient, în special pentru persoanele care au condiții medicale care necesită o supraveghere constantă a semnelor vitale. Într-o eră în care tehnologia joacă un rol esențial în viața cotidiană, utilizarea dispozitivelor mobile și a senzorilor pentru a colecta și analiza date referitoare la sănătatea noastră a devenit din ce în ce mai populară. Aceste sisteme pot oferi o monitorizare continuă a parametrilor fiziologici importanți, cum ar fi pulsul, temperatura corporală, saturația de oxigen din sânge (SpO2) și umiditatea mediului, având un impact semnificativ asupra sănătății pacientului și asupra modului în care pot fi gestionate diferite afecțiuni. Motivul principal pentru alegerea acestui subiect a fost dorința de a contribui la dezvoltarea unor soluții inovative în domeniul telemedicinii, prin crearea unui sistem care să fie ușor de utilizat, precis și eficient, accesibil unui număr cât mai mare de utilizatori, în special celor care au nevoie de o monitorizare constantă.

Lucrarea de față urmărește crearea unui sistem integrat pentru monitorizarea semnelor vitale, care să utilizeze mai mulți senzori pentru măsurarea pulsului, a saturației de oxigen (SpO2), a temperaturii corpului și a umidității mediului înconjurător. Sistemul va fi interconectat cu o aplicație mobilă prin intermediul unui modul Bluetooth, iar datele colectate vor fi prezentate într-o interfață de utilizator prietenoasă și accesibilă. De asemenea, sistemul va include notificări care vor alerta utilizatorul în cazul în care valorile măsurate depășesc limitele normale. Scopul principal este de a oferi o soluție de monitorizare continuă și la distanță, care să fie utilă în prevenirea unor posibile complicații medicale, prin furnizarea de informații în timp real despre starea de sănătate a utilizatorului.

Pentru a realiza acest obiectiv, lucrarea se va concentra pe dezvoltarea și implementarea unui sistem care integrează hardware și software. În partea de hardware, se vor utiliza senzori pentru măsurarea pulsului, a temperaturii și a umidității, precum și un modul Bluetooth pentru transmiterea datelor către o aplicație mobilă. În partea de software, se va dezvolta codul necesar pentru colectarea și prelucrarea datelor de la senzori, pentru gestionarea comunicării Bluetooth și pentru crearea unei interfețe grafice care să permită vizualizarea valorilor în timp real. De asemenea, se va implementa un sistem de notificări care să alerteze utilizatorul în cazul unor valori anormale. În acest context, lucrarea va include și testarea sistemului pentru a evalua performanța și pentru a identifica posibilele îmbunătățiri. Astfel, implementarea unui astfel de sistem va adresa o nevoie reală în domeniul sănătății, având potențialul de a contribui la îmbunătățirea calității vieții

Capitolul 2. Studiul bibliografic și noțiunile teoretice

Capitolul 2 al lucrării se concentrează pe studiul bibliografic și noțiunile teoretice esențiale pentru înțelegerea conceptelor fundamentale care stau la baza dezvoltării sistemului de monitorizare a semnelor vitale. Acest capitol are rolul de a oferi o bază solidă de cunoștințe despre tehnologiile și dispozitivele utilizate în cadrul proiectului, precum și despre principiile teoretice care susțin funcționarea senzorilor și a aplicațiilor de monitorizare. În cadrul acestui capitol, vor fi prezentate informații relevante din literatura de specialitate, care vor ajuta la clarificarea modului în care tehnologiile utilizate în acest proiect contribuie la obținerea unor rezultate precise și fiabile în domeniul monitorizării sănătății.

Studiul bibliografic va include analize ale lucrărilor anterioare care tratează subiecte legate de utilizarea senzorilor pentru măsurarea semnelor vitale, integrarea acestora într-un sistem complex și aplicarea acestora în domeniul telemedicinii. De asemenea, vor fi discutate diversele tehnici și metode de prelucrare a datelor colectate de la senzori, precum și importanța acestora în obținerea unor informații utile și exacte despre starea de sănătate a pacientului. În plus, se va aborda și subiectul aplicațiilor mobile, cu un accent deosebit pe interfețele de utilizator, care joacă un rol esențial în asigurarea unei experiențe ușor de utilizat și eficiente pentru utilizatorii finali. [6]

Noțiunile teoretice discutate în acest capitol vor include principii de funcționare ale senzorilor utilizați în cadrul proiectului, precum senzorii de puls, senzori de temperatură, senzori de umiditate și modulele de comunicare Bluetooth. De asemenea, vor fi explorate concepte esențiale din domeniul programării pentru dispozitivele mobile și al dezvoltării aplicațiilor, inclusiv modalitățile prin care datele provenite de la senzori pot fi transmise, procesate și prezentate într-o aplicație mobilă. Astfel, acest capitol va oferi o înțelegere cuprinzătoare a fundamentelor teoretice necesare pentru dezvoltarea unui sistem eficient de monitorizare a semnelor vitale, punând bazele pentru implementarea practică a acestuia în capitolele următoare ale lucrării. [7]

2.1 Necesitatea tehnologiei IOT

În ultimii ani, dezvoltarea tehnologiilor de tip Internet of Things (IoT) a avut un impact semnificativ asupra diferitelor industrii, iar sectorul medical nu face excepție. IoT în domeniul medical se referă la rețele de dispozitive interconectate care colectează, transmit și procesează date în timp real, având un rol crucial în îmbunătățirea calității îngrijirii pacientului și a eficienței activităților medicale. Necesitatea unui sistem de monitorizare IoT în domeniul medical devine din ce în ce mai evidentă, având în vedere cerințele actuale ale sistemelor de sănătate, creșterea numărului de pacienți cu afecțiuni cronice și nevoia de a oferi îngrijire de calitate într-un mod eficient din punct de vedere al costurilor. [4] Aceste sisteme IoT sunt esențiale pentru furnizarea unor soluții inovative care să răspundă provocărilor actuale din domeniul medical, cum ar fi monitorizarea continuă a pacientului, prevenirea apariției unor complicații și managementul eficient al resurselor. [6]

Unul dintre cele mai importante aspecte ale sistemelor IoT în domeniul medical este capacitatea lor de a permite monitorizarea continuă a stării de sănătate a pacienților, într-un mod non-intruziv și accesibil. [12] De exemplu, pacienții cu boli cronice, cum ar fi diabetul sau hipertensiunea arterială, pot beneficia enorm de pe urma unui astfel de sistem, care le monitorizează semnele vitale, cum ar fi tensiunea arterială, nivelul de zahăr din sânge, temperatura corpului sau ritmul cardiac, și trimite aceste date către medici sau către o aplicație mobilă pentru a fi analizate. Aceste informații în timp real pot ajuta la prevenirea unor eventuale crize sau la intervenirea rapidă în cazul în care starea pacientului se deteriorează. În plus, prin utilizarea acestor sisteme, medicii pot observa evoluția bolii pacientului într-un mod continuu, putând ajusta tratamentele sau recomandările în funcție de nevoile acestuia. [12]

Un alt beneficiu major al sistemelor IoT în domeniul medical este reducerea costurilor și îmbunătățirea eficienței operaționale. Utilizarea unui sistem de monitorizare continuă a pacientului permite medicilor și personalului medical să acorde mai multă atenție pacienților cu risc mare, în timp ce pacienții care sunt într-o stare stabilă pot fi monitorizați de la distanță. Aceasta ajută la economisirea timpului și resurselor, reducând necesitatea unor vizite frecvente la spital sau clinică și permițând o gestionare mai eficientă a resurselor. De asemenea, prin colectarea și analiza datelor, sistemele IoT pot contribui la identificarea timpurie a anumitor tendințe sau a unor anomalii în starea de sănătate a pacienților, ceea ce poate duce la intervenții mai rapide și mai eficiente, prevenind astfel dezvoltarea unor complicații care ar putea necesita tratamente costisitoare și complexe. [11]

Pe lângă acestea, IoT în domeniul medical are un impact semnificativ asupra îmbunătățirii accesului la îngrijire medicală, în special în regiunile izolate sau în zonele cu infrastructură limitată. Multe dintre tehnologiile IoT permit pacienților să își monitorizeze starea de sănătate acasă, fără a fi nevoie să meargă la spital pentru fiecare control. Aceste dispozitive portabile, de exemplu, pot măsura semnele vitale și pot trimite automat datele către un medic, care poate analiza informațiile și poate oferi recomandări. [9] Acest lucru este esențial în zonele rurale sau în regiunile unde accesul la spitale sau clinici este limitat, permițând pacienților să primească îngrijire de calitate fără a fi nevoiți să călătorească pe distanțe lungi. De asemenea, această formă de telemedicină reduce presiunea asupra spitalelor și clinicilor, permițând personalului medical să se concentreze asupra cazurilor mai grave, în timp ce pacienții cu afecțiuni mai ușoare pot beneficia de monitorizare de la distanță. Aceasta contribuie nu doar la îmbunătățirea accesului la îngrijire, dar și la reducerea aglomerației în unitățile medicale, ceea ce este esențial în contextul actualei crize sanitare globale. [11]

Un alt avantaj semnificativ al tehnologiei IoT în sănătate este posibilitatea de a integra diverse tipuri de dispozitive și senzori într-un sistem unic, interconectat, care poate colecta și analiza date din multiple surse. De exemplu, un sistem IoT poate integra senzori pentru monitorizarea ritmului cardiac, a tensiunii arteriale, a temperaturii corpului, a nivelului de oxigen din sânge și chiar a parametrilor de mediu, cum ar fi temperatura și umiditatea. Toate aceste informații pot fi centralizate într-o aplicație mobilă sau într-o platformă online, permițând medicilor și pacienților să acceseze rapid datele și să ia decizii informate. [16] De asemenea, în funcție de datele colectate, sistemul IoT poate alerta automat medicii în cazul în care se detectează vreo anomalie, iar aceștia pot interveni rapid. Această integrare între diferitele dispozitive și tehnologii face ca procesul de monitorizare să fie mult mai eficient și mai precis, contribuind la creșterea siguranței pacienților și la reducerea riscurilor. [12]

Pe lângă aceste beneficii clare, există și provocări importante care trebuie luate în considerare atunci când se implementează sisteme IoT în domeniul medical. Una dintre principalele provocări este asigurarea unui nivel înalt de securitate a datelor. Întrucât sistemele IoT colectează o cantitate mare de informații sensibile despre sănătatea pacientului, protejarea acestor date devine esențială. Orice vulnerabilitate a sistemului ar putea duce la expunerea unor informații personale, ceea ce poate avea consecințe grave, nu doar pentru pacienți, dar și pentru instituțiile medicale. [12] Prin urmare, trebuie implementate măsuri de securitate de ultimă generație, inclusiv criptarea datelor, autentificarea multi-factor și protocoale de comunicație sigure, pentru a proteja confidențialitatea și integritatea informațiilor transmise între dispozitivele IoT și servere. [11]

De asemenea, o altă provocare importantă este asigurarea interoperabilității între diversele dispozitive și platforme. În prezent, există o varietate largă de dispozitive și tehnologii IoT utilizate în domeniul medical, fiecare cu propriile protocoale și specificații. Aceasta poate crea dificultăți în integrarea diferitelor sisteme într-o platformă unificată. De aceea, standardele de interoperabilitate trebuie dezvoltate și adoptate pe scară largă, pentru ca sistemele IoT să poată comunica între ele fără probleme și să ofere date coerente și utile.

2.2 Senzorii utilizați

Senzorii reprezintă componente esențiale în sistemele de monitorizare a parametrilor vitali ai pacientului, având un rol crucial în obținerea de date precise și în timp real despre starea acestora. Utilizarea senzorilor în domeniul medical a evoluat semnificativ în ultimele decenii, iar integrarea acestora în dispozitivele de monitorizare a semnelor vitale a transformat profund modul în care se acordă îngrijirea pacientului. În acest context, senzorii sunt folosiți pentru a colecta informații despre diferite variabile fiziologice, cum ar fi ritmul cardiac, tensiunea arterială, temperatura corpului, nivelul de oxigen din sânge și multe altele, oferind astfel o imagine detaliată și în timp real a stării de sănătate a pacientului. Această tehnologie este crucială în îngrijirea pacienților cu afecțiuni cronice, dar și în prevenirea complicațiilor în cazul celor cu afecțiuni acute sau care se află în post-operatoriu.

Unul dintre cele mai utilizate tipuri de senzori în monitorizarea parametrilor vitali este senzorul de ritm cardiac, care măsoară bătăile inimii ale pacientului pe unitatea de timp, de obicei exprimat în bătăi pe minut (bpm). Acesta poate fi integrat în diverse dispozitive portabile sau fixe, care monitorizează continuu starea pacientului. Senzorii de ritm cardiac funcționează prin detectarea semnalelor electrice emise de inimă în timpul fiecărei bătăi. Aceste semnale sunt captate prin electrozi sau prin alte tehnologii precum fotopletismografia (PPG), care măsoară modificările în volumul de sânge dintr-o arteră. În mod particular, senzorii optici PPG sunt din ce în ce mai utilizați datorită faptului că sunt non-intruzivi și pot fi integrați ușor în dispozitive portabile, cum ar fi ceasurile inteligente sau brățările de fitness. Acești senzori sunt extrem de utili pentru monitorizarea continuă a pacienților cu afecțiuni cardiace, detectând anomalii precum aritmii sau tahicardie, care pot necesita intervenție imediată. [11]

Alți senzori cruciali în monitorizarea semnelor vitale sunt cei care măsoară nivelul de oxigen din sânge (SpO₂). Acest parametru este esențial, deoarece o saturație scăzută a oxigenului din sânge poate duce rapid la complicații severe, cum ar fi insuficiența respiratorie. Senzorii de oxigen utilizează, de obicei, tehnologia de măsurare a absorbției luminii în diferite lungimi de undă, un

principiu similar cu cel utilizat de senzorii de ritm cardiac PPG. Acești senzori sunt adesea integrați în pulsoximetre, care sunt dispozitive portabile ce măsoară în mod constant nivelul de oxigen din sângele periferic. Pulsoximetrele sunt folosite frecvent în spitale pentru monitorizarea pacienților cu afecțiuni respiratorii, dar și acasă, în cazul pacienților care suferă de afecțiuni pulmonare cronice, cum ar fi boala pulmonară obstructivă cronică (BPOC). [5]

În ceea ce privește temperatura corpului, senzorii de temperatură sunt un alt element esențial în monitorizarea stării de sănătate a pacientului. Aceștia măsoară temperatura internă a corpului și pot detecta orice modificare semnificativă, care ar putea semnala prezența unei infecții sau alte condiții patologice. Senzorii de temperatură pot fi utilizați într-o varietate de dispozitive, de la termometre digitale portabile până la dispozitivele mai avansate de monitorizare continuă, care sunt aplicate pe piele sau integrate în echipamentele de spital. [4] Aceste dispozitive pot măsura temperaturi în intervale scurte de timp, ceea ce este extrem de util pentru monitorizarea pacienților în stare gravă sau post-operatorii, unde fluctuațiile temperaturii pot indica complicații imediate.

De asemenea, un alt tip important de senzor utilizat în monitorizarea parametrilor vitali este senzorul de tensiune arterială. Tensiunea arterială este un indicator cheie al sănătății cardiovasculare și este esențială pentru diagnosticarea și gestionarea hipertensiunii arteriale, care este o afecțiune comună și periculoasă. Senzorii de tensiune arterială funcționează prin măsurarea presiunii exercitate de sângele din arterele unei persoane în timpul contracțiilor inimii (presiunea sistolică) și în timpul relaxării inimii (presiunea diastolică). [5] Acești senzori pot fi găsiți într-o varietate de dispozitive, de la monitoare de tensiune portabile până la sisteme de monitorizare continuă care pot detecta modificări subtile ale tensiunii arteriale și care pot ajuta medicii să ajusteze tratamentele în funcție de evoluția pacientului. [5]

În toate aceste cazuri, senzorii sunt integrați într-un sistem mai mare de monitorizare care colectează, transmite și analizează datele pentru a oferi o imagine completă a stării pacientului. În prezent, dispozitivele de monitorizare a parametrilor vitali pot fi conectate la platforme software care permit medicilor să vizualizeze și să analizeze datele în timp real. [12] De asemenea, aceste dispozitive pot fi dotate cu funcționalități de alertă, care trimit notificări către medici sau personalul medical atunci când valorile înregistrate depășesc limitele normale, permițând astfel intervenții rapide și precise. Aceste sisteme sunt din ce în ce mai utilizate în spitale, dar și în aplicațiile de telemedicină, unde pacienții pot fi monitorizați de la distanță, iar datele pot fi transmise direct către medicul curant. [6]

Cu toate acestea, utilizarea senzorilor în monitorizarea parametrilor vitali ridică și câteva provocări, în special în ceea ce privește exactitatea datelor colectate și fiabilitatea acestora în condiții de utilizare extinsă. [12] Unii senzori pot fi influențați de factori externi, cum ar fi temperaturile extreme sau interferențele electromagnetice, ceea ce poate afecta acuratețea măsurătorilor. În plus, problemele legate de gestionarea și protejarea datelor colectate sunt de asemenea importante, având în vedere că informațiile despre sănătatea pacientului sunt extrem de sensibile și trebuie protejate împotriva accesului neautorizat. [11]

În concluzie, senzorii utilizați în dispozitivele de monitorizare a parametrilor vitali sunt elemente esențiale în îmbunătățirea îngrijirii pacientului și în creșterea eficienței sistemelor de sănătate. Aceste tehnologii contribuie la o monitorizare continuă, precisă și non-invazivă a stării de

sănătate a pacienților, oferind atât pacienților, cât și medicilor informații valoroase pentru a preveni complicațiile și a îmbunătăți managementul bolilor. Pe măsură ce tehnologia avansează și noile inovații în domeniul senzorilor sunt implementate, se așteaptă ca aceste dispozitive să devină și mai accesibile și mai eficiente, având un impact pozitiv asupra sănătății globale.

2.3 Avantajele sistemelor de notificare la distanță

Sistemele de alertă la distanță pentru monitorizarea parametrilor vitali ai pacienților reprezintă o revoluție semnificativă în domeniul îngrijirii sănătății, având un impact profund asupra modului în care sunt gestionate afecțiunile cronice, intervențiile medicale și prevenirea complicațiilor. Aceste sisteme, care sunt de obicei integrate în platforme de telemedicină, permit medicilor să monitorizeze continuu starea de sănătate a pacienților, fără a fi nevoie ca aceștia să fie prezenți într-un spital sau să fie conectați la un echipament medical static. Aceste tehnologii au devenit un instrument esențial, în special în contextul creșterii numărului de pacienți cu afecțiuni cronice sau care necesită monitorizare constantă, și au contribuit semnificativ la îmbunătățirea accesului la îngrijire, reducând costurile asociate cu vizitele frecvente la spital și permițând o intervenție rapidă în cazuri de urgență.

Unul dintre principalele avantaje ale sistemelor de alertă pentru monitorizarea la distanță a parametrilor vitali este capacitatea de a colecta și analiza datele în timp real. Pacienții pot fi monitorizați continuu pentru a identifica orice modificare semnificativă a semnelor vitale, cum ar fi ritmul cardiac, tensiunea arterială, nivelul de oxigen din sânge sau temperatura corpului. Aceste modificări, adesea imperceptibile pentru pacient, pot semnaliza posibile complicații sau agravarea unei afecțiuni. [12] De exemplu, un ritm cardiac anormal sau o scădere a nivelului de oxigen poate fi un semnal al unei crize respiratorii, iar o schimbare bruscă a tensiunii arteriale poate indica riscuri legate de accidente vasculare cerebrale. În acest context, sistemele de alertă permit medicilor și personalului medical să intervină rapid, prin notificări automate care sunt trimise către aceștia în momentul în care valorile înregistrate depășesc anumite limite predefinite. Astfel, pacienții beneficiază de un răspuns mai rapid și, de cele mai multe ori, de intervenții care pot salva vieți. [16]

Un alt avantaj semnificativ al acestor sisteme de monitorizare la distanță este că ele reduc necesitatea deplasării fizice a pacientului către unități spitalicești, ceea ce este deosebit de util pentru persoanele cu mobilitate redusă sau pentru cei care locuiesc în zone izolate. [11] De exemplu, pacienții vârstnici sau cei cu dizabilități care nu pot călători ușor pot fi monitorizați constant din confortul propriei case, iar medicii pot interveni imediat atunci când este necesar. Aceasta nu doar că economisește timp și resurse pentru pacient și pentru sistemul de sănătate, dar contribuie și la îmbunătățirea calității vieții pacientului, oferindu-i un sentiment de siguranță și reducând stresul asociat cu vizitele la spital. În plus, monitorizarea la distanță reduce riscurile de infecții nosocomiale, care sunt frecvente în spitale și care pot afecta pacienții vulnerabili sau cei care sunt expuși tratamentelor invazive. [11]

Un alt punct forte al sistemelor de alertă la distanță este faptul că acestea pot fi personalizate în funcție de nevoile și condițiile specifice ale fiecărui pacient. De exemplu, pacienții cu afecțiuni cardiovasculare pot avea parametri specifici care trebuie monitorizați mai frecvent, cum ar fi ritmul cardiac și tensiunea arterială, în timp ce pacienții cu afecțiuni pulmonare pot necesita o monitorizare constantă a nivelului de oxigen din sânge. Sistemele moderne sunt capabile să

adapteze frecvența și sensibilitatea alertelor în funcție de istoricul medical al pacientului, garantând o supraveghere mai atentă și mai eficientă. Această personalizare este esențială pentru prevenirea evenimentelor neașteptate și pentru asigurarea unui tratament adecvat și eficient, întrucât permite detectarea timpurie a oricăror semne de agravare a stării pacientului. [11]

Un alt avantaj considerabil al monitorizării de la distanță este capacitatea de a analiza datele colectate pe o perioadă mai lungă de timp, ceea ce ajută la identificarea unor modele și tendințe care nu ar fi fost evidente prin măsurători ocazionale. [11] De exemplu, fluctuațiile ritmului cardiac pe termen lung sau variațiile nivelului de oxigen pot semnaliza condiții care nu sunt vizibile în cadrul unui control punctual. Această analiză de lungă durată poate ajuta medicii să anticipeze eventuale complicații sau să ajusteze tratamentele într-un mod mai informativ și mai precis. Mai mult decât atât, aceste date pot fi stocate și utilizate pentru a crea statistici de sănătate la nivel de populație, contribuind astfel la cercetarea medicală și la îmbunătățirea generală a serviciilor de sănătate. [13]

Cu toate acestea, implementarea acestor tehnologii nu este lipsită de provocări. Unul dintre principalele obstacole întâmpinate în utilizarea sistemelor de alertă la distanță este legat de confidențialitatea și securitatea datelor pacienților. Informațiile referitoare la sănătatea unei persoane sunt extrem de sensibile și trebuie protejate cu cele mai înalte standarde de securitate pentru a preveni accesul neautorizat. În plus, există riscul ca unele date să fie interpretate greșit sau ca alertele să fie generate de erori tehnice, ceea ce ar putea conduce la intervenții nejustificate sau, dimpotrivă, la ignorarea unor semne care necesită atenție urgentă. Pentru a aborda aceste provocări, este necesar ca sistemele de monitorizare să fie dotate cu tehnologii avansate de criptare și autentificare, iar personalul medical să fie instruit corespunzător în interpretarea datelor și în gestionarea alertelor. [15]

Pe lângă aceste provocări tehnice, există și aspecte legate de accesibilitatea și adaptabilitatea acestor sisteme, mai ales pentru pacienții din zone rurale sau pentru cei care nu sunt familiarizați cu tehnologia. De asemenea, pentru a fi cu adevărat eficiente, aceste sisteme trebuie să fie integrate într-un ecosistem de sănătate care include atât infrastructura necesară (conectivitate de internet stabilă, dispozitive compatibile etc.), cât și un cadru legislativ care să reglementeze utilizarea și protecția datelor. [11]

În concluzie, sistemele de alertă la distanță pentru monitorizarea parametrilor vitali reprezintă un pas important în modernizarea îngrijirii pacientului și în eficientizarea proceselor din domeniul medical. Acestea permit o supraveghere continuă și adaptată nevoilor individuale ale fiecărui pacient, contribuind la îmbunătățirea calității vieții și la prevenirea unor complicații grave. Cu toate că există provocări legate de securitatea datelor, accesibilitatea tehnologică și integrarea acestora în sistemele de sănătate existente, avantajele acestor tehnologii sunt clare și merită investiții semnificative pentru dezvoltarea și implementarea lor la scară largă. Implementarea acestora poate revoluționa modul în care pacienții sunt îngrijiți, reducând presiunea asupra sistemelor de sănătate tradiționale și oferind o îngrijire mai rapidă, mai precisă și mai accesibilă pentru toți pacienții. [11]

2.4 Sisteme similare existente pe piața

Analizarea sistemelor similare cu cel dezvoltat în această lucrare este esențială pentru înțelegerea contextului în care inovăm și pentru identificarea celor mai bune practici care pot fi integrate în propria soluție. Aceste studii comparative permit nu doar învățarea din succesul altora, dar și recunoașterea eventualelor lacune sau provocări care pot apărea în dezvoltarea unui astfel de sistem. În plus, analiza pieței și a sistemelor existente ajută la evaluarea fezabilității proiectului, oferind o viziune clară asupra concurenței și a cerințelor tehnice necesare. De asemenea, aceste comparații pot sprijini procesul de optimizare a tehnologiilor, deoarece soluțiile existente oferă o bază solidă pe care se pot construi noi funcționalități sau îmbunătățiri. În continuare, voi analiza trei exemple de sisteme reale care sunt similare cu soluția de monitorizare a parametrilor vitali a pacienților dezvoltate în această lucrare, astfel încât să putem înțelege mai bine impactul și aplicabilitatea acestui tip de tehnologie.

Un exemplu notabil este sistemul de monitorizare a sănătății dezvoltat de compania **Fitbit**. Produsele Fitbit sunt renumite pentru capacitatea lor de a urmări o gamă largă de parametri vitali, de la ritmul cardiac, numărul de pași, până la calitatea somnului și nivelul de activitate fizică al utilizatorului. De asemenea, dispozitivele Fitbit pot fi conectate la aplicații mobile, unde utilizatorii pot vizualiza datele în timp real și pot primi rapoarte detaliate privind starea lor de sănătate. Un aspect inovator al acestor dispozitive este integrarea cu aplicații de sănătate precum Google Fit și Apple Health, oferind utilizatorilor o experiență integrată și completă. Deși Fitbit nu se concentrează exclusiv pe monitorizarea pacienților cu afecțiuni medicale grave, produsele sale au fost utilizate pentru prevenirea unor afecțiuni, îmbunătățirea calității vieții și menținerea unui stil de viață sănătos. Comparativ cu soluția propusă în această lucrare, Fitbit oferă o experiență mai generalistă, concentrându-se pe utilizatorii sănătoși, în timp ce soluția de monitorizare a parametrilor vitali din lucrarea noastră este mai specific orientată către pacienții care necesită monitorizare constantă din cauza unor afecțiuni cronice.

Un alt exemplu relevant este **Withings**, o companie care oferă soluții avansate pentru monitorizarea sănătății. Produsele lor includ brățări inteligente, termometre, dispozitive de monitorizare a tensiunii arteriale și cântare inteligente. Cu ajutorul dispozitivelor **Withings**, pacienții și utilizatorii pot monitoriza și urmări parametrii vitali precum tensiunea arterială, ritmul cardiac, temperatura corpului și chiar analiza compoziției corpului. Aceste dispozitive sunt însoțite de aplicații mobile care permit utilizatorilor să își vizualizeze datele într-un mod clar și accesibil, iar unele dintre aceste dispozitive pot fi integrate în sistemele de telemedicină, permițând medicilor să monitorizeze starea de sănătate a pacienților de la distanță. Un avantaj major al produselor Withings este capacitatea lor de a integra datele colectate cu istoricul medical, ceea ce oferă medicilor o viziune mai clară asupra evoluției pacientului. În comparație cu sistemul dezvoltat în această lucrare, Withings se concentrează mai mult pe dispozitivele de uz general și pe integrarea acestora în viața cotidiană a utilizatorilor, în timp ce soluția noastră este destinată mai mult monitorizării continue și urgente a pacienților cu afecțiuni grave, având astfel o aplicabilitate diferită și o abordare mai dedicată unui segment specific de utilizatori.

Un al treilea exemplu semnificativ în domeniul monitorizării sănătății este **Masimo**, o companie recunoscută la nivel global pentru tehnologia sa de monitorizare non-invazivă a semnelor vitale, în special pentru SpO₂, pulsul arterial și alți parametri cardio-respiratori. Masimo oferă soluții tehnologice avansate care sunt folosite în spitale, clinici și pentru monitorizarea pacienților de

acasă. Produsele lor sunt extrem de precise și sunt folosite pentru monitorizarea pacienților în timp real, dar și pentru măsurători pe termen lung, esențiale în gestionarea afecțiunilor respiratorii și cardiace. Masimo a dezvoltat tehnologia *Signal Extraction Technology* (SET), care permite măsurarea precisă a nivelului de oxigen în sânge, chiar și în condiții dificile, cum ar fi pacienții cu mișcări sau cei cu un flux sanguin scăzut. Aceasta este o tehnologie esențială în spital și este folosită pentru a asigura monitorizarea continuă a pacienților, fără a fi nevoie de proceduri invazive. În comparație cu soluția de monitorizare a parametrilor vitali propusă în această lucrare, Masimo se concentrează pe medii clinice și spitale, fiind mai puțin orientat spre telemedicină și monitorizarea la distanță a pacienților, în timp ce soluția noastră vizează integrarea unui sistem complet de monitorizare a sănătății pentru pacienți, chiar și pentru cei care sunt departe de instituțiile medicale.

Fiecare dintre aceste exemple – Fitbit, Withings și Masimo – demonstrează importanța și aplicabilitatea sistemelor de monitorizare a sănătății în viața cotidiană, dar și în mediul clinic. Fiecare companie adresează nevoia de a urmări și analiza parametrii vitali într-un mod diferit, având avantaje și limitări care le fac potrivite pentru diverse aplicații. Comparând aceste soluții cu sistemul propus în lucrarea noastră, observăm că, în timp ce tehnologiile existente se concentrează pe utilizatorii de zi cu zi sau pe pacienții din spitale, soluția propusă este mai adaptată pentru monitorizarea continuă și alertarea rapidă a medicilor în cazul pacienților cu afecțiuni cronice sau periculoase. De asemenea, este important de menționat că tehnologiile existente sunt deja bine integrate în ecosistemele de sănătate și oferă o bază solidă pe care soluțiile inovative, cum este cea din lucrarea noastră, pot construi pentru a oferi o îngrijire mai eficientă și accesibilă tuturor.

2.5 Probleme ale sistemelor existente și oportunități de dezvoltare

În analiza sistemelor existente, am identificat mai multe probleme și limitări care afectează performanța acestora în contextul monitorizării parametrilor vitali ai pacienților. Aceste probleme sunt evidente în produsele exemplificate anterior, precum Fitbit, Withings și Masimo, și sunt legate de precizia măsurărilor, complexitatea implementării, costurile ridicate ale echipamentelor, integrarea dificilă în sistemele medicale de urgență și, nu în ultimul rând, limitele tehnologiei de monitorizare la distanță. Aceste dificultăți pot compromite eficiența unui sistem de monitorizare continuă, în special în cazul pacienților care suferă de afecțiuni grave sau care necesită o supraveghere constantă, motiv pentru care soluția propusă în această lucrare vine să adreseze aceste deficiențe și să ofere un sistem mai accesibil și mai eficient în îngrijirea pacienților.

Una dintre principalele probleme întâlnite la sistemele de monitorizare existente este legată de precizia măsurărilor, mai ales în condiții de utilizare intensă sau în medii mai dificile. De exemplu, produsele Fitbit și Withings, care sunt destinate utilizatorilor din afaceri de sănătate generală sau celor care doresc să își monitorizeze starea de bine, pot să nu fie suficient de precise atunci când sunt utilizate în cazuri medicale severe. Aceste dispozitive au fost concepute pentru a măsura parametrii vitali într-un mod simplificat și generalizat, dar pentru pacienții care suferă de afecțiuni respiratorii, cardiace sau alte boli grave, cerințele sunt mult mai ridicate. De asemenea, aceste dispozitive pot fi influențate de factori externi, cum ar fi mișcarea bruscă a pacientului sau interferențele electromagnetice, ceea ce poate duce la erori în măsurători.

În schimb, soluția noastră propusă în această lucrare se concentrează pe precizia și fiabilitatea datelor colectate. Am utilizat tehnologii avansate pentru a minimiza erorile de măsurare, inclusiv senzorii de puls și temperatura corpului, care au fost calibrați în prealabil și care sunt conectați la un sistem de monitorizare continuă ce garantează măsurători exacte și constante, indiferent de condițiile externe. De asemenea, am integrat notificări și alerte automate care sunt transmise în timp real, permițând intervenția rapidă în cazul în care valorile parametrilor vitali ale pacientului ies din intervalele normale.

O altă problemă comună întâlnită la sistemele de monitorizare deja existente este complexitatea implementării și costurile ridicate ale echipamentelor. De exemplu, sistemele de monitorizare dezvoltate de Masimo sunt extrem de precise, dar sunt destinate în principal instituțiilor medicale mari, având un preț considerabil și fiind necesare echipamente suplimentare pentru integrarea în infrastructuri deja existente. De asemenea, soluțiile de acest tip nu sunt accesibile pentru toți pacienții, în special pentru cei din zone izolate sau din regiunile cu resurse financiare limitate.

Pentru a rezolva aceste probleme, soluția noastră a fost construită pe un principiu de accesibilitate și costuri reduse. Am dezvoltat un sistem bazat pe hardware și software deschise, utilizând tehnologii disponibile pe piață la prețuri rezonabile. De asemenea, am simplificat instalarea și utilizarea echipamentelor, astfel încât pacienții să poată monitoriza parametrii vitali într-un mod ușor de implementat, fără necesitatea unor infrastructuri costisitoare sau complexe. Sistemul nostru permite monitorizarea continuă a pacientului, fiind adaptabil pentru utilizarea în mediul de acasă, dar și în spitale mai mici sau clinici cu resurse limitate.

Integrarea în sistemele de urgență reprezintă un alt punct slab al sistemelor existente. Deși unele produse, precum cele oferite de Withings, permit integrarea cu aplicații mobile de sănătate, procesul de alerte și notificări poate fi adesea limitat. Aceste sisteme nu sunt suficient de rapide și de eficiente în a transmite informațiile de urgență către medici, în special în cazuri critice, când timpul de reacție este esențial. În cazul produselor Fitbit și Withings, utilizatorii nu beneficiază întotdeauna de o interacțiune directă cu sistemele de urgență, iar datele colectate sunt adesea analizate doar de utilizatorul final și nu sunt transmise într-un mod eficient către profesioniștii din domeniul sănătății.

Sistemul nostru dezvoltat în această lucrare depășește aceste limitări prin integrarea în timp real a datelor în sistemele de urgență. Când un pacient depășește un anumit prag de parametri vitali (de exemplu, pulsul sau temperatura corporală), o alertă automată este trimisă direct la o echipă medicală specializată, care poate interveni rapid pentru a oferi asistență. Această integrare rapidă a informațiilor reduce semnificativ timpul de reacție al echipelor de urgență, ceea ce poate face diferența între viață și moarte în cazuri critice. În plus, datele sunt colectate și procesate într-un sistem centralizat, unde medicii pot vizualiza istoricul pacientului și pot lua decizii informate bazate pe informații exacte și complete.

În final, soluția propusă în această lucrare adresează o serie de deficiențe esențiale întâlnite în sistemele de monitorizare existente, iar prin integrarea unui sistem complet, accesibil și eficient, putem asigura un nivel mai ridicat de siguranță pentru pacienți. Problemele legate de precizia măsurărilor, costurile ridicate și integrarea cu sistemele de urgență au fost depășite prin tehnologia aleasă, iar inovațiile aduse pot contribui semnificativ la îmbunătățirea îngrijirii

pacienților, în special a celor care necesită monitorizare continuă. Astfel, sistemul nostru oferă nu doar o soluție tehnologică, ci și o abordare nouă în managementul sănătății, ce poate salva vieți în momente critice.

Capitolul 3. Prezentarea sistemului propus spre dezvoltare și componentele folosite

3.1 Prezentarea sistemului propus spre dezvoltare

Această lucrare își propune să dezvolte un sistem modern de monitorizare a parametrilor vitali ai pacienților, destinat utilizării atât în mediile clinice, cât și la domiciliu. Prin combinarea unor tehnologii accesibile cu un design inovator, proiectul își propune să furnizeze un instrument de monitorizare precis și ușor de utilizat. Sistemul este conceput pentru a colecta, prelucra și analiza date referitoare la starea de sănătate a pacienților, oferind în același timp alerte în timp real atunci când valorile parametrilor monitorizați depășesc limitele prestabilite. Dezvoltarea acestui sistem are ca scop principal sprijinirea personalului medical și a utilizatorilor finali prin furnizarea unor date vitale esențiale în mod rapid și eficient, ceea ce contribuie la luarea deciziilor informate într-un timp scurt. Conceptul sistemului integrează o platformă hardware compactă și eficientă, senzori pentru măsurarea semnelor vitale și a condițiilor ambientale, un modul de afișare locală a datelor și o aplicație mobilă pentru analiza și stocarea informațiilor, asigurând astfel un flux de lucru coerent și fiabil.

Caracteristicile principale ale sistemului

- Măsurarea ritmului cardiac și a saturației oxigenului (SpO2) utilizând un senzor MAX30100.
- Monitorizarea temperaturii corporale printr-un senzor de tip NTC.
- Determinarea condițiilor ambientale, precum temperatura și umiditatea relativă, utilizând un senzor SHT11.
- Prezentarea datelor în timp real pe un ecran LCD, oferind utilizatorilor o interfață locală ușor de citit.
- Transmiterea datelor către o aplicație mobilă Android prin intermediul unui modul Bluetooth HC-05, cu notificări automate pentru valori anormale.

Subsistemele proiectului

Subsistemul de achiziție a datelor

Subsistemul de achiziție a datelor este responsabil pentru colectarea informațiilor brute de la senzori și asigurarea integrității acestora înainte de prelucrare. Senzorul MAX30100 este utilizat

pentru măsurarea ritmului cardiac și a saturației oxigenului din sânge (SpO₂), utilizând o metodă optică bazată pe absorbția luminii. Acesta integrează LED-uri de culoare roșie și infraroșie, precum și un fotodetector sensibil, care detectează variațiile fluxului sanguin. Datele sunt procesate inițial prin algoritmi specifici care filtrează zgomotul și elimină artefactele cauzate de mișcare sau alte interferențe. Senzorul NTC este destinat măsurării temperaturii corporale prin intermediul unei rezistențe care își schimbă valoarea în funcție de temperatură. Pentru a asigura precizia datelor, acest senzor necesită o calibrare atentă, având în vedere factorii externi precum variațiile de temperatură ambientală. Pe de altă parte, senzorul SHT11 este un modul digital compact care măsoară atât temperatura, cât și umiditatea relativă a aerului. Acest senzor utilizează un convertor analog-digital intern pentru a furniza valori precise, iar ieșirea sa digitală este ideală pentru aplicații medicale. Datele brute colectate de la toți acești senzori sunt transmise către microcontrolerul principal pentru prelucrare ulterioară.

Subsistemul de procesare și afișare locală

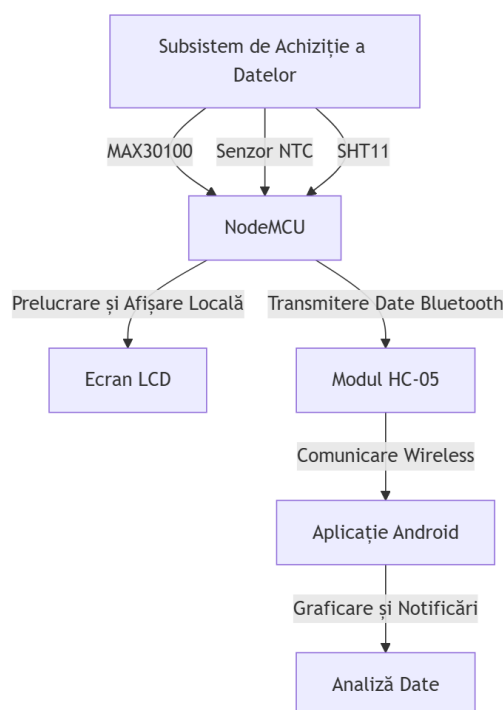
Subsistemul de procesare și afișare locală are ca scop principal interpretarea datelor colectate de la senzori și prezentarea lor într-un format accesibil utilizatorului. Acest subsistem este centrat pe placa de dezvoltare NodeMCU, care utilizează un microcontroler ESP8266. Acest microcontroler este responsabil pentru primirea datelor de la senzori prin intermediul interfețelor analogice și digitale, precum și pentru aplicarea unor algoritmi de prelucrare pentru eliminarea zgomotului și extragerea valorilor utile. Odată procesate, datele sunt afișate pe un ecran LCD de tip 16x2 sau 20x4, care utilizează o interfață I2C pentru a reduce numărul de pini necesari. Afișajul local permite utilizatorului să vizualizeze în timp real valorile parametrilor vitali, oferind o soluție rapidă și accesibilă pentru monitorizare. Implementarea software-ului pentru acest subsistem implică utilizarea unor biblioteci dedicate pentru fiecare senzor, precum și dezvoltarea unor algoritmi personalizați pentru actualizarea eficientă a datelor afișate.

Subsistemul de comunicare și analiză pe platforma mobilă

Ultimul subsistem al sistemului se concentrează pe comunicarea wireless și analiza datelor pe o platformă mobilă. Acesta utilizează un modul Bluetooth HC-05 pentru a transmite datele prelucrate de la microcontroler către o aplicație Android personalizată. Modulul Bluetooth este configurat pentru a comunica cu microcontrolerul printr-o interfață UART, iar datele sunt transmise într-un format serial care poate fi interpretat de aplicația mobilă. Aplicația Android a fost dezvoltată pentru a oferi o interfață intuitivă și prietenoasă, permițând utilizatorilor să vizualizeze datele sub formă de grafice și tabele în timp real. În plus, aplicația include un sistem de notificări automate care alertează utilizatorul atunci când valorile parametrilor vitali depășesc limitele normale. Aceasta integrează funcționalități suplimentare, precum stocarea istoricului datelor și generarea de rapoarte personalizate, asigurând astfel o monitorizare continuă și detaliată a stării de sănătate a pacientului.

3.2 Schema bloc a sistemului

Diagrama bloc realizată reprezintă structura generală a sistemului de monitorizare a parametrilor vitali și evidențiază principalele componente și legături funcționale dintre acestea. Sistemul este împărțit în mai multe subsisteme, fiecare având un rol bine definit în procesul de achiziție, prelucrare, transmitere și analiză a datelor.



Prima componentă principală a diagramei este Subsistemul de Achiziție a Datelor, care integrează senzorii utilizați pentru măsurarea parametrilor vitali și a condițiilor ambientale. Senzorul MAX30100 colectează informații despre ritmul cardiac și saturația oxigenului (SpO2) folosind tehnologia optică. Senzorul NTC este responsabil pentru măsurarea temperaturii corporale prin variații de rezistență electrică în funcție de temperatură. De asemenea, senzorul SHT11 determină temperatura și umiditatea ambientală, furnizând date esențiale pentru o monitorizare completă. Toate aceste componente transmit datele brute către microcontrolerul principal, reprezentat de NodeMCU.

NodeMCU este componenta centrală a sistemului, fiind responsabilă pentru preluarea datelor de la senzori și prelucrarea acestora. Diagrama ilustrează cum informațiile sunt transmise din subsistemul de achiziție către microcontroler, unde sunt aplicate algoritmi specifici pentru a elimina zgomotul și pentru a asigura precizia măsurătorilor. NodeMCU gestionează și afișarea locală a datelor prin intermediul unui ecran LCD conectat. Această componentă permite utilizatorului să vizualizeze în timp real valorile măsurate, oferind o interfață simplă și intuitivă pentru monitorizare.

Fig 3.1 Schema bloc a sistemului

Un alt aspect important al diagramei este legătura dintre NodeMCU și Modulul Bluetooth HC-05, care asigură transmiterea datelor procesate către aplicația mobilă. Modulul Bluetooth este configurat pentru a comunica cu microcontrolerul printr-un protocol serial, iar datele sunt trimise în timp real către un dispozitiv Android. Acest proces asigură o conexiune stabilă și rapidă, fiind esențial pentru monitorizarea la distanță.

În continuare, Aplicația Android este responsabilă pentru preluarea și analiza datelor transmise de modulul Bluetooth. Aplicația este configurată pentru a afișa valorile parametrilor sub formă de grafice și pentru a genera notificări automate atunci când sunt detectate anomalii. Aceasta oferă utilizatorului o perspectivă clară asupra stării de sănătate a pacientului, cu posibilitatea de a stoca istoricul măsurărilor și de a genera rapoarte detaliate.

În concluzie, diagrama bloc evidențiază fluxul de date și interacțiunile dintre componentele sistemului. Fiecare bloc din diagramă are un rol bine definit, contribuind la funcționarea generală a sistemului și la atingerea obiectivelor acestuia: monitorizarea rapidă, precisă și accesibilă a parametrilor vitali ai pacienților.

3.3 Costul de implementare

Implementarea unui astfel de sistem de monitorizare a parametrilor vitali implică o analiză detaliată a costurilor pentru a se asigura că acesta rămâne accesibil, dar totodată eficient și funcțional. Costurile de implementare joacă un rol esențial, mai ales în contextul în care sistemul ar putea fi utilizat în aplicații personale sau în mediul medical, unde optimizarea raportului calitate-preț este esențială. Un sistem bine proiectat nu doar că oferă performanțe ridicate, dar își justifică și investiția prin accesibilitatea tehnologiei și ușurința în utilizare.

Primul element central al sistemului este placa de dezvoltare **NodeMCU**, care funcționează ca unitate principală de procesare și control. Aceasta integrează un microcontroler ESP8266, capabil să gestioneze atât intrările și ieșirile senzorilor, cât și transmiterea datelor prin Bluetooth. Costul mediu al unei plăci NodeMCU pe piața locală este de aproximativ **40 RON**, ceea ce o face o alegere accesibilă și versatilă pentru proiecte de acest tip.

Un alt element critic este **senzorul MAX30100**, utilizat pentru măsurarea ritmului cardiac și a nivelului de oxigen din sânge. Acest senzor, datorită complexității sale tehnice și preciziei ridicate, are un cost mediu de **50 RON**. Prețul său este justificat de performanța superioară în monitorizarea parametrilor vitali, fiind o componentă indispensabilă pentru acest proiect.

Pentru măsurarea temperaturii corporale, sistemul folosește un **senzor NTC**. Acesta este o soluție simplă și accesibilă, cu un cost mediu de aproximativ **10 RON**. Deși necesită calibrare atentă pentru a asigura precizia măsurărilor, senzorul NTC reprezintă o opțiune rentabilă pentru monitorizarea temperaturii.

De asemenea, în cadrul sistemului este inclus **senzorul SHT11**, utilizat pentru măsurarea temperaturii și umidității ambientale. Acesta este un senzor de înaltă performanță, cu ieșire digitală, iar costul său mediu este de **70 RON**. Prețul reflectă fiabilitatea și precizia oferite, fiind un element important pentru monitorizarea condițiilor externe care pot influența starea de sănătate a pacientului.

Pentru afișarea datelor în timp real, este utilizat un **ecran LCD**, de tip 16x2 sau 20x4, conectat prin interfață I2C. Acest tip de afișaj are un cost mediu de **40 RON**, iar dimensiunea și claritatea sa îl fac potrivit pentru o interfață locală ușor de citit.

Un alt element esențial al sistemului este **modulul Bluetooth HC-05**, care asigură transmiterea datelor către aplicația mobilă. Acesta are un cost mediu de **30 RON** și oferă o conexiune stabilă, fiind ideal pentru implementarea unei soluții de monitorizare la distanță.

În plus, este necesar să se țină cont de costurile asociate cu elementele auxiliare, cum ar fi cablurile, firele de conectare, rezistențele, condensatorii și șasiul de protecție al sistemului. Aceste componente suplimentare sunt esențiale pentru integrarea și protejarea sistemului și adaugă un cost estimativ de **80 RON**.

Cost total estimat: 320 RON

Acest cost total face ca sistemul să fie o soluție accesibilă, având în vedere complexitatea sa și beneficiile oferite. Importanța acestui cost derivă din faptul că oferă un punct de plecare clar pentru bugetare, permițând, de asemenea, personalizarea sistemului prin înlocuirea anumitor componente cu alternative mai avansate sau mai accesibile, în funcție de cerințele utilizatorului.

3.4 Componentele folosite

În cadrul acestui capitol, vor fi prezentate în detaliu componentele utilizate pentru implementarea sistemului de monitorizare a parametrilor vitali. Alegerea fiecărei componente a fost realizată pe baza cerințelor tehnice și funcționale ale proiectului, având în vedere factori precum precizia, eficiența, compatibilitatea și costul. Fiecare element contribuie la asigurarea unui sistem robust și fiabil, capabil să îndeplinească obiectivele propuse. De asemenea, vor fi evidențiate caracteristicile principale ale fiecărei componente și rolul specific al acesteia în cadrul sistemului.

3.4.1 Placa de dezvoltare

Placa de dezvoltare reprezintă componenta centrală a acestui sistem, având rolul de a coordona toate funcțiile dispozitivului, de la colectarea datelor de la senzori până la transmiterea acestora către aplicația mobilă și afișarea locală pe ecranul LCD. Alegerea unei plăci de dezvoltare în locul realizării unui circuit imprimat personalizat (PCB) cu un microcontroler dedicat s-a dovedit a fi o decizie practică și eficientă. Plăcile de dezvoltare sunt soluții pre-asamblate, optimizate și testate pentru a oferi stabilitate și performanță, eliminând astfel necesitatea unui proces complex de proiectare și fabricare a PCB-ului. În cazul unui proiect de tip prototip, cum este acesta, utilizarea unei plăci de dezvoltare economisește timp și resurse, oferind totodată flexibilitate pentru modificări și ajustări ulterioare.

Realizarea unui PCB personalizat implică un proces extins, care include selectarea și dimensionarea componentelor electronice, realizarea designului pentru traseele electrice, fabricarea propriu-zisă a plăcii și asamblarea componentelor. Pe lângă aceste etape, se adaugă și riscul erorilor de proiectare, care pot necesita modificări costisitoare și consumatoare de timp. În schimb, o placă de dezvoltare precum NodeMCU sau Arduino Uno vine cu toate circuitele și funcțiile de bază integrate, oferind o soluție rapidă și sigură pentru implementarea unui sistem complex.

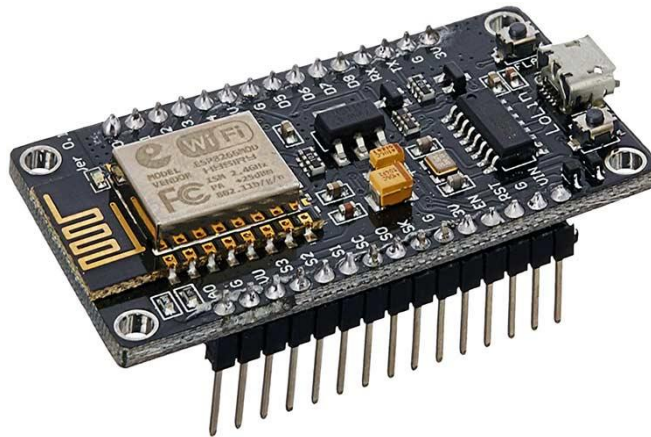


Fig 3.2 Placa de dezvoltare

În cadrul acest proiect, au fost luate în considerare două opțiuni pentru placa de dezvoltare: **Arduino Uno** și **NodeMCU ESP8266**. Arduino Uno este una dintre cele mai populare plăci de dezvoltare, fiind apreciată pentru simplitatea sa și pentru suportul extins oferit de comunitatea de utilizatori. Aceasta integrează un microcontroler ATmega328P și oferă un număr suficient de pini digitali și analogici pentru majoritatea proiectelor. Cu toate acestea, una dintre limitările majore ale plăcii Arduino Uno este lipsa unor capabilități de conectivitate wireless integrate, ceea ce ar fi necesitat adăugarea unui modul Bluetooth extern. De asemenea, memoria RAM și puterea de procesare a Arduino Uno sunt limitate comparativ cu NodeMCU, ceea ce poate deveni o constrângere în cazul unui proiect care necesită prelucrarea mai multor seturi de date în timp real.

Pe de altă parte, NodeMCU ESP8266 oferă un set de avantaje care au justificat alegerea sa pentru acest proiect. NodeMCU este o placă de dezvoltare compactă și puternică, bazată pe microcontrolerul ESP8266, care integrează atât un procesor pe 32 de biți, cât și o conectivitate Wi-Fi. Deși acest proiect nu utilizează conectivitatea Wi-Fi, ESP8266 oferă performanțe superioare în ceea ce privește viteza de procesare, memoria RAM disponibilă și flexibilitatea generală a platformei. NodeMCU include, de asemenea, un regulator de tensiune și o interfață USB integrată pentru programare și alimentare, ceea ce simplifică semnificativ utilizarea în aplicații practice.

Istoria NodeMCU și a microcontrolerului ESP8266 este una remarcabilă, ilustrând dezvoltarea rapidă a platformelor IoT accesibile și performante. Microcontrolerul ESP8266, dezvoltat de compania Espressif Systems în 2014, a fost inițial conceput ca un modul Wi-Fi low-cost pentru proiectele electronice. Datorită caracteristicilor sale avansate și prețului redus, ESP8266 a devenit rapid popular în rândul dezvoltatorilor. Ulterior, platforma NodeMCU a fost creată pentru a facilita utilizarea ESP8266, oferind o placă de dezvoltare ușor de utilizat și un firmware bazat pe limbajul de programare Lua. În timp, NodeMCU a evoluat, devenind una dintre cele mai versatile soluții pentru proiecte IoT și embedded.

În concluzie, alegerea plăcii NodeMCU ESP8266 pentru acest proiect a fost determinată de performanțele sale superioare, costul accesibil și flexibilitatea platformei. Deși conectivitatea Wi-Fi nu este utilizată în cadrul acestui sistem, capacitățile avansate ale ESP8266 asigură că

placa poate gestiona cu ușurință sarcinile de procesare și comunicare necesare. Această decizie reflectă un echilibru optim între performanță, cost și ușurința de implementare, asigurând succesul proiectului.

3.4.2 Senzorul de temperatură și umiditate

În cadrul acestui proiect, monitorizarea parametrilor de mediu, cum ar fi temperatura și umiditatea, este o componentă esențială pentru o înțelegere completă a condițiilor în care se află pacientul. Senzorul responsabil pentru această funcție este **DHT11**, o soluție compactă și accesibilă, recunoscută pentru fiabilitatea sa în măsurarea acestor parametri. Alegerea DHT11 s-a bazat pe o analiză detaliată a opțiunilor disponibile și pe evaluarea cerințelor proiectului, care au dus la concluzia că acesta este cea mai potrivită soluție.

În procesul de selecție, au fost analizate două opțiuni: **SHT31** și **DHT11**. Senzorul SHT31 este o soluție avansată, care oferă o precizie superioară și o gamă mai largă de temperaturi și niveluri de umiditate măsurate. SHT31 este un senzor digital bazat pe tehnologia CMOSens, ce permite o ieșire rapidă și precisă a datelor, fiind utilizat frecvent în aplicații industriale și medicale. Cu toate acestea, utilizarea senzorului SHT31 în acest proiect a fost limitată de prețul său ridicat, care poate ajunge la peste dublul costului unui DHT11. Într-un context în care precizia absolută a datelor nu este critică, iar bugetul este o constrângere importantă, DHT11 reprezintă o alternativă excelentă, menținând un echilibru între cost și performanță.

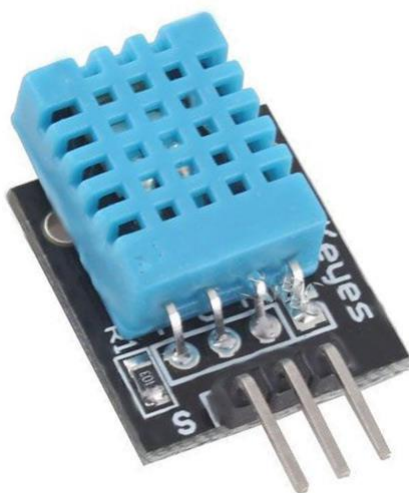


Fig 3.3 Senzorul DHT11

Pe de altă parte, senzorul DHT11 este o soluție simplă și rentabilă, care își îndeplinește cu succes rolul în acest proiect. Acesta oferă o măsurare suficient de precisă a temperaturii și umidității pentru aplicațiile de monitorizare personală, fiind ideal pentru prototipuri și proiecte care nu necesită standarde extrem de stricte. În plus, DHT11 este ușor de utilizat datorită interfeței sale digitale simple, ceea ce simplifică procesul de integrare în sistem. Alegerea senzorului DHT11 a fost motivată de avantajele sale economice, de simplitatea în utilizare și de disponibilitatea largă pe piață, toate acestea făcându-l o opțiune excelentă pentru acest proiect.

Senzorul DHT11 funcționează pe baza unui termistor pentru măsurarea temperaturii și a unui senzor capacitiv pentru determinarea umidității. Termistorul este un dispozitiv sensibil la temperatură, care modifică rezistența electrică în funcție de variațiile de temperatură, iar senzorul capacitiv detectează schimbările de umiditate pe baza variațiilor capacității dintre două plăci condensatoare. Aceste semnale analogice sunt convertite în date digitale de un microcontroler intern integrat în senzor, iar acestea sunt transmise către microcontrolerul principal al sistemului printr-o interfață digitală simplă. Acest proces asigură măsurători rapide și fiabile, cu o precizie suficientă pentru aplicațiile generale.

Din punct de vedere istoric, DHT11 a fost dezvoltat ca o soluție accesibilă pentru măsurarea parametrilor de mediu, fiind utilizat inițial în proiecte educaționale și aplicații de tip DIY. De-a lungul timpului, datorită fiabilității sale, DHT11 a devenit o componentă populară în proiectele de automatizare și IoT. Acesta oferă o gamă de măsurare a temperaturii între 0°C și 50°C, cu o precizie de $\pm 2^\circ\text{C}$, și o gamă de măsurare a umidității relative între 20% și 90%, cu o precizie de $\pm 5\%$. Deși limitele sale sunt mai restrânse în comparație cu senzorii mai avansați, cum ar fi SHT31, performanța sa este mai mult decât adecvată pentru aplicații casnice și educaționale.

Capabilitățile senzorului DHT11 îl fac ideal pentru acest proiect, oferind o soluție compactă, cu un consum energetic redus și o interfață simplă de utilizat. În plus, compatibilitatea sa cu microcontrolere precum NodeMCU asigură o integrare fără probleme în sistem, permițând obținerea rapidă și eficientă a datelor de mediu. Alegerea acestui senzor reflectă o balanță bine calculată între cost, performanță și cerințele specifice ale proiectului, demonstrând că soluțiile accesibile pot oferi rezultate remarcabile atunci când sunt utilizate corespunzător.

3.4.3 Senzorul de puls

Senzorul **MAX30100** joacă un rol esențial în cadrul acestui proiect, fiind responsabil pentru măsurarea ritmului cardiac și a nivelului de oxigen din sânge (SpO_2). Aceste două valori sunt printre cei mai importanți indicatori ai stării de sănătate a unui pacient, oferind informații critice despre funcționarea sistemului cardiovascular și despre nivelul de oxigenare a organismului. Alegerea acestui senzor s-a bazat pe cerințele specifice ale proiectului, inclusiv precizia măsurătorilor, integrarea ușoară și eficiența energetică. În procesul de selecție, două opțiuni au fost analizate: un senzor IR analogic și MAX30100, fiecare având puncte forte și limitări specifice.

Senzorul IR analogic reprezintă o soluție simplă și accesibilă pentru măsurarea ritmului cardiac, utilizând un LED cu infraroșu pentru a detecta variațiile fluxului sanguin în funcție de bătăile inimii. Principiul său de funcționare se bazează pe măsurarea variațiilor de intensitate a luminii reflectate sau transmise printr-un țesut (de obicei, degetul sau lobul urechii). Deși această soluție este rentabilă și poate fi utilizată cu succes în aplicații mai puțin pretențioase, senzorul IR analogic are limitări semnificative. Printre acestea se numără sensibilitatea redusă la variații minore ale fluxului sanguin, imposibilitatea de a măsura saturația de oxigen și necesitatea unui circuit analogic suplimentar pentru procesarea semnalului, ceea ce complică integrarea în proiecte mai complexe.

În schimb, MAX30100 este un senzor digital avansat care oferă funcționalități integrate pentru măsurarea atât a ritmului cardiac, cât și a nivelului de SpO_2 . Acesta combină într-un singur

modul un LED roșu, un LED infraroșu, un fotodetector și un convertor analog-digital (ADC) de înaltă precizie. Prin emiterea luminii roșii și infraroșii către țesut și măsurarea cantității de lumină absorbită și reflectată, senzorul poate calcula cu acuratețe atât frecvența cardiacă, cât și saturația de oxigen din sânge. Această tehnologie duală permite o măsurare mai completă și mai precisă, eliminând nevoia de circuite externe complexe și reducând semnificativ dimensiunea și consumul energetic al soluției.

Din punct de vedere istoric, senzorul MAX30100 a fost dezvoltat de **Maxim Integrated** (acum parte a Analog Devices), un lider global în domeniul tehnologiei semiconductoarelor și al circuitelor integrate analogice. Lansat în jurul anului 2015, MAX30100 a fost conceput special pentru aplicații portabile și dispozitive de monitorizare a sănătății, cum ar fi brățările de fitness, dispozitivele de monitorizare a pulsului și sistemele de monitorizare medicală. Acesta a fost unul dintre primii senzori integrați care au combinat măsurători ale ritmului cardiac și ale SpO₂ într-un singur modul compact, stabilind un nou standard pentru tehnologia senzorilor biometrici.



Fig 3.4 Senzorul de puls

Capabilitățile senzorului MAX30100 sunt remarcabile, având specificații care îl fac ideal pentru o gamă largă de aplicații. Acesta este capabil să măsoare frecvențe cardiace între 30 și 240 de bătăi pe minut (BPM) și niveluri de SpO₂ între 70% și 100%, cu o precizie ridicată. Senzorul include, de asemenea, funcții de compensare a zgomotului și a interferențelor optice, ceea ce asigură măsurători fiabile chiar și în condiții de iluminare variabilă. În plus, consumul său energetic este extrem de redus, ceea ce îl face ideal pentru dispozitive portabile alimentate de baterii.

Alegerea senzorului MAX30100 pentru acest proiect a fost determinată de superioritatea sa tehnologică față de soluțiile analogice tradiționale. În timp ce senzorii IR analogici pot oferi doar o măsurătoare de bază a ritmului cardiac, MAX30100 oferă o soluție completă, care adaugă valoare semnificativă sistemului prin măsurarea precisă a SpO₂. În plus, integrarea ușoară a senzorului datorită interfeței sale I2C și a designului compact a simplificat procesul de dezvoltare, reducând numărul de componente externe necesare.

În concluzie, senzorul MAX30100 reprezintă o alegere excelentă pentru acest proiect, datorită capacităților sale avansate, istoriei sale dovedite de performanță și fiabilitate și suportului extins oferit de producător. Prin integrarea acestuia, sistemul devine capabil să ofere date precise și

relevante despre starea de sănătate a pacientului, consolidându-și astfel utilitatea și valoarea în contextul monitorizării parametrilor vitali.

3.4.4 Senzorul pentru temperatura corpului

În cadrul acestui proiect de monitorizare a sănătății, măsurarea temperaturii corpului este un parametru esențial, având un impact semnificativ asupra diagnosticului și tratamentului pacientului. Temperatura corpului este un indicator important al stării de sănătate, iar fluctuațiile sale pot semnaliza diverse afecțiuni, de la febră, care poate indica o infecție, până la hipotermie. Pentru a realiza această măsurare, în proiect a fost ales un **senzor de temperatură NTC** (Negative Temperature Coefficient), un tip de termistor care își modifică rezistența pe măsură ce temperatura variază. Acest senzor NTC, parte integrantă a sistemului, joacă un rol crucial în obținerea unei imagini complete asupra stării pacientului, fiind responsabil pentru măsurarea continuă a temperaturii corpului.

Pentru a obține date precise de temperatură folosind un senzor NTC, acesta trebuie integrat într-un circuit de tip **divizor de tensiune**, care transformă variațiile rezistenței senzorului într-un semnal de tensiune măsurabil. Principiul de funcționare al unui senzor NTC este simplu: pe măsură ce temperatura crește, rezistența senzorului scade. Acesta este tipul de comportament caracteristic unui termistor NTC. Într-un divizor de tensiune, senzorul NTC este plasat într-un circuit format dintr-o rezistență cunoscută și o sursă de alimentare stabilă. Tensiunea măsurată în punctul de mijloc al divizorului de tensiune variază în funcție de temperatura măsurată de senzor, iar această tensiune poate fi ulterior convertită într-o valoare de temperatură exactă utilizând un algoritm de conversie specific. Această metodă este simplă și eficientă, având un cost scăzut și o precizie adecvată pentru scopul proiectului.

În cadrul acestui proiect, senzorul NTC utilizat este modelul **B3435**, un senzor de temperatură popular datorită caracteristicilor sale tehnice și fiabilității. Modelul B3435 este un termistor de tip NTC, cu un coeficient negativ de temperatură specific, care îi permite să ofere măsurători precise pe o gamă largă de temperaturi. Acesta are o gamă de operare cuprinsă între -40°C și 125°C , ceea ce îl face potrivit pentru o mare varietate de aplicații, inclusiv pentru monitorizarea temperaturii corpului uman. Deși nu este la fel de precis ca senzorii de temperatură de tip digital, precum cei din familia DS18B20, B3435 oferă un raport calitate-preț excelent, având un cost semnificativ mai scăzut și fiind ușor de integrat într-un circuit analogic simplu.

Istoria senzorului B3435 este legată de evoluția tehnologiei termistorilor, care au fost dezvoltate pentru a îmbunătăți măsurarea temperaturii în diverse aplicații electronice. Termistorii NTC sunt folosiți de decenii în industrie și în proiectele DIY datorită performanței lor stabile și a capacității de a oferi măsurători precise într-o gamă largă de temperaturi. De-a lungul timpului, aceștia au fost utilizați în aplicații care variază de la controlul temperaturii în echipamente electronice la monitorizarea condițiilor de mediu. Modelul B3435 este fabricat de mai mulți producători de componente electronice și este disponibil pe scară largă, ceea ce îl face ușor de procurat și utilizat pentru proiecte de mică amploare și prototipurile de tip hobby.



Fig 3.5 Senzorul de temperatură

Capabilitățile senzorului B3435 includ o rezistență nominală de $10k\Omega$ la 25°C , cu o variație liniară și predictibilă a rezistenței în funcție de temperatură, ceea ce îl face ușor de calibrare și de utilizat pentru obținerea unui semnal de temperatură stabil. De asemenea, acest senzor are o sensibilitate mai mare în comparație cu alți senzori de temperatură, precum termocuplurile sau senzorii RTD (Resistive Temperature Detector), iar în combinație cu un divizor de tensiune, poate oferi măsurători precise și fiabile. În plus, B3435 este ușor de integrat cu microcontrolerele, precum NodeMCU, și nu necesită componente suplimentare costisitoare pentru prelucrarea semnalului.

Alegerea senzorului B3435 pentru acest proiect se bazează pe mai mulți factori, inclusiv costul redus, ușurința în integrare și stabilitatea măsurătorilor sale. Într-un sistem în care precizia extremă a temperaturii nu este critică, B3435 oferă o soluție eficientă și economică, care satisface pe deplin cerințele proiectului. În plus, funcționarea sa stabilă și fiabilă îl face potrivit pentru o aplicație de monitorizare continuă, în care temperatura corpului este un parametru important pentru evaluarea stării pacientului.

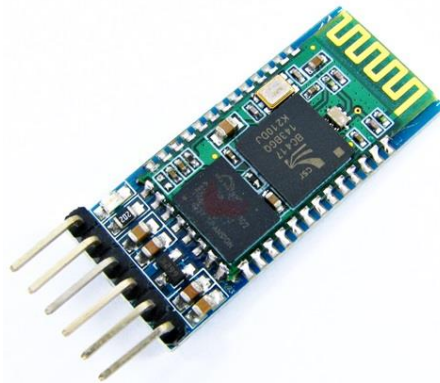
În concluzie, senzorul de temperatură NTC B3435 reprezintă o alegere excelentă datorită caracteristicilor sale de performanță și a costului scăzut. Acesta se integrează perfect în sistemul de monitorizare, furnizând date precise despre temperatura corpului, care sunt esențiale pentru evaluarea stării de sănătate a pacientului. Alegerea B3435 reflectă o decizie calculată, care ia în considerare atât cerințele tehnice, cât și limitările bugetare ale proiectului, asigurând astfel succesul implementării acestuia.

3.4.5 Modulul bluetooth

În cadrul acestui proiect, modulul HC-05 joacă un rol esențial în asigurarea conectivității wireless între sistemul de monitorizare a pacientului și aplicația mobilă personalizată. Acesta permite transmiterea datelor în timp real, precum ritmul cardiac, nivelul de oxigen din sânge, temperatura corpului și alți parametri, către dispozitivele mobile, oferind o modalitate

convenabilă și eficientă de acces la informațiile relevante. Alegerea modului HC-05 a fost dictată de cerințele specifice ale proiectului, care impuneau o soluție wireless de comunicare de înaltă performanță, dar și accesibilă din punct de vedere al costului. În această secțiune, vom explora cum funcționează modulul HC-05, istoricul său, capacitățile tehnice și motivele pentru care a fost ales pentru acest proiect.

Modulul HC-05 este un modul de comunicație Bluetooth care permite microcontrolerelor să stabilească o conexiune wireless cu dispozitive externe, cum ar fi telefoanele mobile, tabletele sau alte unități Bluetooth. În cadrul acestui sistem de monitorizare a pacientului, HC-05 este folosit pentru a trimite datele măsurate de senzorii de temperatură, ritm cardiac și oxigenare a sângelui către aplicația mobilă personalizată. Astfel, pacientul sau utilizatorul poate vizualiza în timp real evoluția stării de sănătate pe un smartphone sau o tabletă. De asemenea, HC-05 poate fi folosit pentru a primi comenzi sau instrucțiuni de la aplicația mobilă, cum ar fi modificarea



intervalelor de monitorizare sau setarea alertei pentru parametrii din afara limitelor.

Fig 3.6 Modulul HC-05

Modulul HC-05 funcționează pe banda de frecvență 2.4 GHz, care este standardul pentru majoritatea dispozitivelor Bluetooth. Acesta poate fi setat pentru a funcționa în mod master sau slave, în funcție de necesitățile aplicației. În mod master, HC-05 poate iniția și controla conexiuni, în timp ce în mod slave, acesta poate accepta conexiuni de la un dispozitiv master (de exemplu, un smartphone). Această versatilitate a făcut din HC-05 o alegere potrivită pentru acest proiect, în care comunicarea între modulul de monitorizare și dispozitivele mobile este un aspect fundamental.

Modulul HC-05 funcționează pe principiul standardelor de comunicație Bluetooth, care permit transmiterea datelor prin undă radio pe distanțe scurte, de obicei până la 10 metri, în condiții ideale. Pentru a stabili o conexiune între HC-05 și dispozitivul mobil, utilizatorul trebuie să inițieze căutarea dispozitivelor Bluetooth, iar HC-05 va apărea în lista de dispozitive disponibile. După stabilirea conexiunii, datele pot fi transferate rapid și eficient între microcontrolerul care colectează informațiile de la senzori și aplicația mobilă care le va prelucra și afișa în grafice.

Modulul HC-05 este compatibil cu protocoalele standard seriale, iar comunicarea se face printr-o interfață UART (Universal Asynchronous Receiver Transmitter), ceea ce înseamnă că datele sunt transmise sub formă de biți asincroni printr-un port serial al microcontrolerului (cum ar fi NodeMCU). Acest proces este rapid și eficient, iar implementarea sa nu necesită componente hardware suplimentare complexe. Modulul poate fi configurat și controlat prin comenzi AT (AT Commands), care permit configurarea parametrilor de comunicație, cum ar fi numele dispozitivului, viteza de transfer a datelor și multe altele.

Un alt aspect important al funcționării modulului HC-05 este gestionarea consumului de energie. Deși Bluetooth-ul, în general, poate consuma mai multă energie comparativ cu alte tehnologii de comunicare wireless, HC-05 a fost conceput pentru a funcționa eficient în termeni de consum energetic. Acest lucru este un avantaj semnificativ pentru aplicațiile de monitorizare a sănătății, în care economisirea energiei este crucială pentru a asigura o durată de viață lungă a bateriei dispozitivului portabil.

Alegerea modulului HC-05 pentru acest proiect a fost influențată de mai mulți factori tehnici și economici. În primul rând, HC-05 este extrem de ușor de utilizat și de integrat cu microcontrolere populare, cum ar fi NodeMCU sau Arduino. Configurarea sa simplă prin comenzi AT și interfața serială face ca integrarea să fie rapidă și eficientă, economisind timp și resurse de dezvoltare. De asemenea, costul redus al modulului HC-05 comparativ cu alte soluții de comunicare wireless, cum ar fi modulele Wi-Fi sau modulele Zigbee, a făcut ca acesta să fie o alegere economică excelentă pentru acest proiect, având în vedere bugetul limitat și obiectivele de performanță ale proiectului.

Un alt factor important care a influențat alegerea HC-05 este compatibilitatea acestuia cu majoritatea telefoanelor mobile moderne, care includ Bluetooth ca tehnologie standard. Acest lucru permite o conectivitate ușoară și rapidă între sistemul de monitorizare și aplicația mobilă, fără a fi necesare hardware-uri suplimentare sau adaptori de conectivitate. De asemenea, modulul HC-05 este versatil, având posibilitatea de a fi configurat atât pentru comunicații de tip master, cât și slave, ceea ce îl face extrem de flexibil în implementarea diferitelor scenarii de comunicație.

Modulul HC-05 a fost lansat de HC Technology și este unul dintre cele mai populare module Bluetooth utilizate în proiectele electronice de mică amploare și prototipurile de tip DIY. De-a lungul anilor, HC-05 a devenit un standard de facto în ceea ce privește modulele Bluetooth pentru microcontrolere, datorită fiabilității și performanței sale. De asemenea, HC-05 a fost utilizat într-o gamă largă de aplicații, de la controlul de la distanță al dispozitivelor electronice la transferul de date wireless între diferite sisteme.

Printre capabilitățile tehnice ale HC-05 se numără transferul de date la viteză mare (până la 3 Mbps), comunicarea bidirecțională, și posibilitatea de a opera la distanțe de până la 10 metri, în condiții ideale. În plus, modulul include un regulator de tensiune intern, ceea ce înseamnă că poate funcționa în mod fiabil cu o gamă largă de tensiuni de alimentare (de obicei între 3.3V și 6V), ceea ce îl face compatibil cu majoritatea platformelor de dezvoltare electronice.

În concluzie, modulul HC-05 este alegerea ideală pentru acest proiect datorită performanței sale fiabile, ușurinței de utilizare și costului redus. Prin integrarea sa, proiectul beneficiază de o soluție wireless eficientă și flexibilă, care permite monitorizarea în timp real a parametrilor pacientului și transmiterea rapidă a acestora către aplicația mobilă. Capacitățile de configurare, împreună cu compatibilitatea largă și costurile accesibile, fac din HC-05 un component indispensabil în realizarea unui sistem de monitorizare a sănătății portabil și funcțional.

3.4.6 Ecranul LCD

În cadrul sistemului de monitorizare a pacientului, ecranul LCD 20x4 joacă un rol crucial în oferirea unei interfețe vizuale pentru utilizator, permițându-i să vizualizeze în timp real datele colectate de la senzori, cum ar fi ritmul cardiac, nivelul de oxigen din sânge și temperatura corpului. Alegerea unui astfel de ecran pentru proiect a fost dictată de necesitatea de a furniza o soluție accesibilă și ușor de integrat, care să permită o vizualizare clară și rapidă a datelor. În această secțiune, vom explora de ce a fost ales un LCD 20x4, ce alte opțiuni au fost luate în considerare și care sunt caracteristicile tehnice și istoricul acestui tip de ecran.

Ecranul LCD 20x4 este esențial pentru a oferi utilizatorilor o interfață de vizualizare a datelor într-un format ușor de citit. Acesta poate afișa până la 20 de caractere pe fiecare din cele patru linii ale ecranului, ceea ce permite prezentarea a patru parametri simultan, fiecare pe câte o linie diferită. În cadrul acestui proiect, LCD-ul 20x4 este folosit pentru a afișa parametrii monitorizați, cum ar fi temperatura corpului, ritmul cardiac, nivelul de oxigen din sânge și eventual alte mesaje de stare sau alerte. Acesta oferă o soluție accesibilă, intuitivă și eficientă pentru afișarea datelor, fără a necesita o aplicație mobilă sau un alt dispozitiv extern pentru vizualizare.

Un alt avantaj al LCD-ului 20x4 este faptul că acesta poate fi citit în orice condiții de iluminare, comparativ cu alte tipuri de ecrane care ar putea necesita un iluminat suplimentar. De asemenea, este ușor de integrat cu microcontrolere, precum NodeMCU sau Arduino, având interfețe standard (precum parallel sau I2C), care permit transmiterea datelor către ecran rapid și eficient.

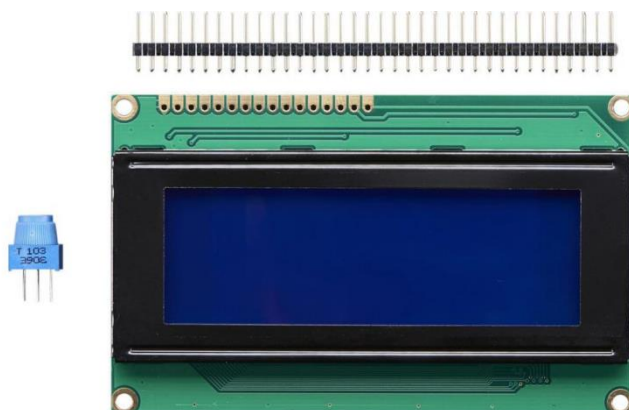
Înainte de a lua decizia finală de a utiliza un LCD 20x4, au fost luate în considerare și alte opțiuni pentru afișarea datelor, inclusiv ecranele OLED. Ecranele OLED oferă avantajul unui contrast mult mai înalt și al unui consum mai mic de energie în comparație cu LCD-urile tradiționale. În plus, OLED-urile au o gamă mai largă de culori și un unghi de vizualizare mai larg, ceea ce le face atractive pentru multe aplicații. Totuși, în cadrul acestui proiect, am ales să utilizăm un LCD 20x4 din mai multe motive.

Primul motiv este costul. Ecranele OLED sunt, de obicei, mai scumpe decât LCD-urile tradiționale, iar în contextul unui proiect cu un buget limitat, LCD-ul 20x4 oferă o soluție mult mai economică. De asemenea, ecranele LCD 20x4 sunt larg disponibile și pot fi găsite într-o gamă variată de producători, ceea ce face alegerea și achiziția mai ușoare.

Al doilea motiv se referă la complexitatea implementării. LCD-urile 20x4 au o documentație mai largă și sunt mai bine susținute în comunitățile de dezvoltatori, ceea ce face integrarea lor într-un sistem bazat pe microcontroler mult mai rapidă și mai simplă. Pe de altă parte, ecranele OLED,

deși sunt interesante din punct de vedere tehnic, pot necesita o configurare mai detaliată și o gestionare mai complexă a resurselor.

În concluzie, LCD-ul 20x4 a fost ales pentru acest proiect datorită costului accesibil, ușurinței de implementare și disponibilității largi, fiind soluția optimă pentru a prezenta informațiile într-un



mod clar și eficient.

Fig 3.7 Ecranul LCD

Ecranele LCD 20x4 fac parte din familia de ecrane cu cristale lichide și sunt printre cele mai utilizate tipuri de ecrane pentru aplicațiile electronice de mică amploare și prototipurile DIY. De-a lungul decadelor, LCD-urile au evoluat semnificativ în ceea ce privește tehnologia și performanța lor. Deși au apărut inițial în dispozitivele de mare consum, cum ar fi calculatoarele portabile și ceasurile electronice, LCD-urile s-au impus și în domeniul microcontrolerelor datorită costului scăzut, fiabilității și capacității de a fi integrate ușor în diverse tipuri de circuite.

În ceea ce privește capacitățile tehnice ale LCD-ului 20x4, acesta oferă un spațiu suficient pentru a vizualiza patru linii de câte 20 de caractere fiecare, ceea ce este ideal pentru afișarea unui număr redus de parametri, fără a copleși utilizatorul cu prea multe informații. De asemenea, LCD-urile sunt capabile să afișeze caractere alfanumerice, precum și simboluri speciale, ceea ce le face foarte versatile pentru diverse aplicații. Aceste ecrane sunt alimentate de obicei la 5V, iar interfața de comunicație poate fi realizată prin porturi paralel sau seriale, în funcție de tipul de driver utilizat.

În cadrul acestui proiect, LCD-ul 20x4 este echipat cu un driver I2C, care permite o comunicare mult mai eficientă cu microcontrolerul. Driverul I2C (Inter-Integrated Circuit) este un protocol de comunicație care permite transferul de date între microcontroler și LCD folosind doar două linii: una pentru semnalul de ceas (SCL) și alta pentru date (SDA). Aceasta este o alegere foarte convenabilă, deoarece economisește pini de pe microcontroler, lăsându-i disponibili pentru alte funcționalități.

Un alt avantaj al utilizării unui driver I2C pentru LCD este reducerea complexității wiring-ului. Fără I2C, LCD-ul 20x4 ar necesita un număr mare de linii de conexiune (de obicei 6 sau 7 pini),

ceea ce ar putea complica circuitul. Cu driverul I2C, toate conexiunile necesare pentru controlul LCD-ului se fac pe doar două linii, ceea ce face circuitul mult mai curat și mai ușor de realizat.

Driverul I2C permite, de asemenea, controlul ușor al luminozității și contrastului ecranului prin software, iar comenzi precum clear display sau cursor movement pot fi executate rapid și eficient. În plus, utilizarea unui driver I2C pentru LCD contribuie la reducerea consumului de energie, ceea ce este esențial pentru un sistem de monitorizare de lungă durată, alimentat de baterii.

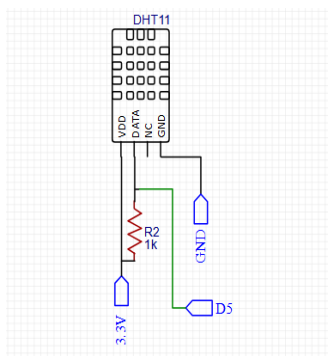
În concluzie, alegerea unui LCD 20x4 echipat cu un driver I2C a fost o decizie bazată pe mai mulți factori, inclusiv costul, simplitatea implementării și eficiența comunicației. LCD-ul 20x4 oferă o soluție robustă și clară pentru afișarea datelor colectate de la senzorii proiectului, iar driverul I2C simplifică foarte mult integrarea sa într-un sistem bazat pe microcontroler. Deși ecranele OLED ar fi fost o alternativă interesantă, LCD-ul 20x4 s-a dovedit a fi cea mai bună alegere datorită raportului excelent între performanță și cost.

Capitolul 4. Dezvoltarea componentei hardware a sistemului

În acest capitol, vom explora procesul de dezvoltare a componentei hardware a sistemului de monitorizare a pacientului, abordând în detaliu alegerea și integrarea fiecărui component electric esențial pentru funcționarea corespunzătoare a întregului sistem. Fiecare component hardware, de la senzorii de măsurare a parametrilor biologici la modulele de comunicare wireless, joacă un rol crucial în colectarea datelor și transmiterea acestora către dispozitivele mobile ale utilizatorilor.

Pentru a asigura o implementare corectă și eficientă, fiecare componentă hardware a fost detaliată printr-un schematic electric propriu. Aceste scheme electrice reprezintă diagrama conexiunilor individuale ale fiecărui element, oferind o reprezentare clară și exactă a modului în care componentele sunt interconectate. Fiecare componentă a fost conectată cu atenție, iar schema electrică relevantă a fost realizată pentru a reflecta precis traseele de semnal, alimentarea și interfețele dintre piese.

Prin urmare, acest capitol va detalia nu doar alegerea fiecărui component, dar și cum au fost realizate conexiunile electrice și de ce fiecare dintre acestea a fost aleasă pentru a contribui la performanța optimă a sistemului. Fiecare schema electrică va fi însoțită de o explicație detaliată, astfel încât cititorul să poată înțelege logica din spatele fiecărei decizii de proiectare hardware,



dar și interacțiunea dintre componente.

Fig 4.1 Schema electrică a senzorului DHT

Primul componentă hardware este senzorul DHT11, un senzor utilizat pentru măsurarea temperaturii și umidității din mediul înconjurător. Acest senzor are patru pini: VCC, GND, DATA și un pin neutilizat în cazul nostru. VCC este conectat la sursa de alimentare de 3.3V, iar GND este conectat la masă, ceea ce asigură alimentarea corespunzătoare și funcționarea senzorului. Conexiunea dintre DATA și pinul de control al microcontrolerului este realizată prin intermediul unui rezistor de 1kΩ, care joacă un rol esențial în gestionarea semnalului de comunicare între senzor și microcontroler.

În ceea ce privește rolul fiecărei conexiuni, VCC, fiind conectat la 3.3V, furnizează tensiunea necesară pentru funcționarea senzorului DHT11. De obicei, acești senzori pot funcționa la o gamă de tensiuni între 3.3V și 5V, însă pentru a proteja componentele microcontrolerului și pentru a se asigura că senzorul nu depășește limita de tensiune tolerabilă, 3.3V este o alegere optimă. Conexiunea între GND și pinul de masă al microcontrolerului este un pas obligatoriu pentru a închide circuitul și a asigura un punct de referință de tensiune corect. În ceea ce privește pinul de DATA, acesta este folosit pentru transmiterea datelor de la senzor către microcontroler.

Una dintre cele mai importante componente ale acestui circuit este rezistorul de 1kΩ plasat între VCC și pinul de date. Rolul acestui rezistor este de a proteja semnalul de transmisie și de a ajusta nivelul semnalului. În cadrul acestui senzor, semnalul de ieșire de pe pinul de date poate fluctua între niveluri logice înalte și joase. Rezistorul de 1kΩ este plasat pentru a limita curentul care circulă prin linia de date și pentru a ajuta la menținerea unui semnal stabil și fiabil, prevenind suprasarcinile care ar putea deteriora componentele electronice. Mai precis, acesta previne fluctuațiile bruște de tensiune care ar putea apărea în timpul comunicării cu microcontrolerul, asigurând o integrare lină a senzorului în sistem.

La intersecția dintre rezistor și pinul de date, am conectat pinul D5 al microcontrolerului. Pinul D5 este responsabil pentru citirea datelor de la senzorul DHT11 și trimiterea acestora către procesorul microcontrolerului, unde vor fi interpretate și procesate. Această legătură între senzor și microcontroler permite transmiterea informațiilor de temperatură și umiditate, care vor fi ulterior afișate pe ecranul LCD sau trimise către aplicația mobilă prin modulul Bluetooth HC-05.

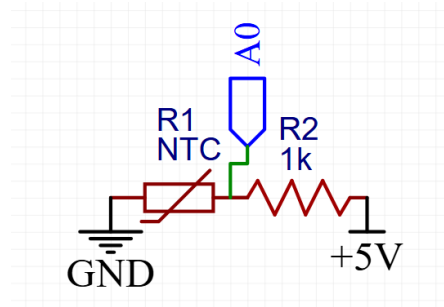
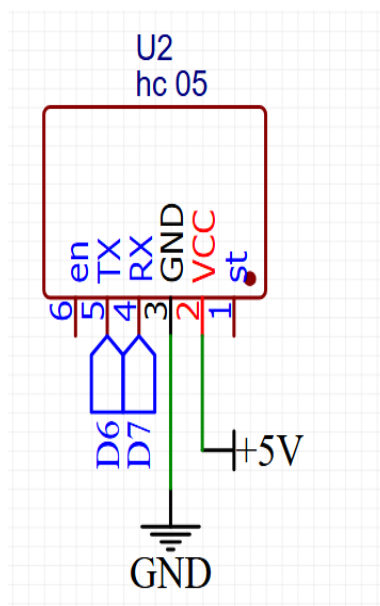


Fig 4.2 Schema electrică a senzorului NTC

În cadrul sistemului, al doilea senzor utilizat pentru măsurarea temperaturii este un senzor NTC (Negative Temperature Coefficient), care își schimbă rezistența în funcție de temperatura la care este expus. Acesta joacă un rol important în monitorizarea temperaturii corpului pacientului. În acest caz, senzorul NTC este conectat într-un circuit denumit divizor de tensiune împreună cu un rezistor de 10k Ω , pentru a putea măsura variațiile rezistenței și pentru a o transforma într-un semnal analogic pe care microcontrolerul îl poate citi.

În acest circuit, o parte din senzorul NTC este conectată la GND, iar cealaltă parte este plasată într-un circuit serie cu rezistorul de 10k Ω . Cealaltă extremitate a rezistorului de 10k Ω este, de asemenea, conectată la GND. Astfel, circuitul creează o configurație de divizor de tensiune, care permite măsurarea unei tensiuni analogice care variază în funcție de temperatura mediului în care se află senzorul.



Un divizor de tensiune este un circuit simplu format din două rezistoare conectate în serie. Tensiunea de intrare este aplicată la capătul superior al primului rezistor, iar tensiunea de ieșire este luată de la punctul de joncțiune dintre cei doi rezistori. În acest caz, senzorul NTC acționează ca unul dintre rezistori, iar celălalt este un rezistor fix de $10k\Omega$. Tensiunea aplicată pe acest divizor este împărțită între cei doi rezistori, iar punctul de ieșire va depinde de rezistența senzorului NTC, care variază în funcție de temperatura măsurată. De fapt, valoarea rezistenței NTC scade pe măsură ce temperatura crește, ceea ce face ca tensiunea la punctul de joncțiune să varieze.

În acest circuit, pinul A0 al microcontrolerului este conectat la joncțiunea dintre cei doi rezistori, acolo unde tensiunea divizată este măsurată. Acest pin analogic va citi tensiunea generată de divizorul de tensiune, iar valoarea acesteia va fi folosită pentru a calcula temperatura, pe baza caracteristicilor senzorului NTC. Cu cât temperatura crește, cu atât rezistența senzorului NTC scade, iar tensiunea citită pe pinul A0 va reflecta acest comportament, ceea ce permite procesorului microcontrolerului să determine temperatura mediului.

Astfel, voltajul divizat joacă un rol crucial în acest sistem, pentru că este transforma în semnal analogic și este folosit de microcontroler pentru a calcula temperatura. Circuitul de divizor de tensiune funcționează în mod eficient pentru a adapta semnalul de la senzorul NTC, care are o rezistență variabilă, într-un semnal de tensiune constantă ce poate fi citit de un pin analogic al microcontrolerului. Aceasta este o metodă simplă și eficientă de a măsura parametrii analogici într-un sistem bazat pe microcontrolere.

Fig 4.3 Schema electrică a modului Bluetooth

În cadrul proiectului, modulul HC-05 este utilizat pentru comunicarea wireless cu aplicația mobilă. Acesta este un modul Bluetooth ce permite transmiterea datelor între microcontroler și un dispozitiv mobil. Conexiunile modului HC-05 sunt simple, dar esențiale pentru a asigura o comunicație corectă și eficientă. Vom discuta despre fiecare conexiune în parte și despre cum funcționează comunicarea prin intermediul porturilor seriale.

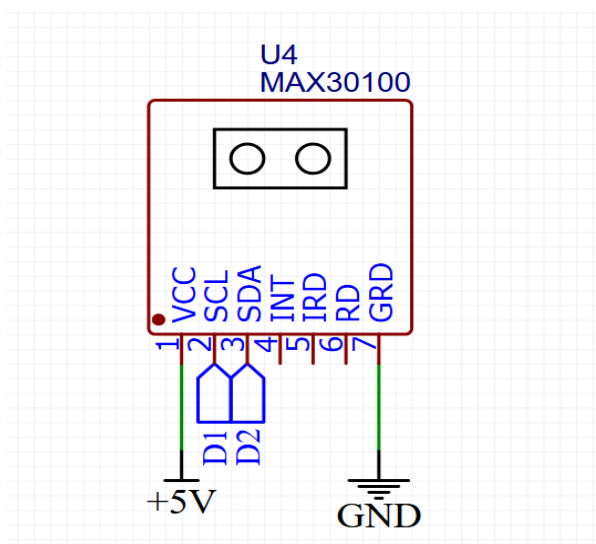
Primul pin al modului HC-05 este VCC, care este conectat la o sursă de alimentare de 5V. Modulul HC-05 necesită o tensiune de 3.6V până la 6V pentru a funcționa corect, iar conexiunea la 5V asigură alimentarea necesară pentru a face posibilă comunicația Bluetooth. Al doilea pin, GND, este conectat la masa circuitului pentru a închide circuitul și a oferi referință la nivelul de tensiune de 0V. Aceste două conexiuni sunt esențiale pentru alimentarea corectă a modului și pentru a asigura o funcționare stabilă a acestuia.

Modulul HC-05 include două pini pentru comunicația serială: RX și TX, care sunt responsabili pentru transmiterea și primirea datelor. Pinul RX al HC-05 este conectat la pinul D6 al microcontrolerului, iar pinul TX al HC-05 este conectat la pinul D7 al microcontrolerului. Aceste conexiuni sunt realizate pentru a folosi portul serial software al microcontrolerului, ceea ce înseamnă că, deși microcontrolerul are un port serial hardware (USART), se utilizează un port serial virtual creat prin software pentru a comunica cu modulul Bluetooth. Acest lucru permite

microcontrolerului să transmită și să primească date prin intermediul modulului HC-05 fără a utiliza portul serial hardware, care poate fi rezervat pentru alte sarcini.

Comunicarea între microcontroler și modulul HC-05 se realizează printr-un protocol de comunicație serială (UART - Universal Asynchronous Receiver-Transmitter). În acest caz, datele sunt trimise de la microcontroler la HC-05 prin pinul TX al microcontrolerului și pinul RX al HC-05, iar datele primite de HC-05 sunt trimise înapoi către microcontroler prin pinul TX al HC-05 și pinul RX al microcontrolerului. Datorită utilizării portului serial software, microcontrolerul poate configura orice pin disponibil pentru a funcționa ca RX sau TX, ceea ce adaugă flexibilitate în designul sistemului.

Comunicarea wireless se bazează pe protocoale de tip Bluetooth care permit schimbul de date între două dispozitive aflate în apropiere. Modulul HC-05 este configurat să comunice cu un telefon mobil sau un alt dispozitiv Bluetooth compatibil. Atunci când microcontrolerul transmite date către modulul HC-05, acesta le trimite prin protocolul Bluetooth la aplicația mobilă, iar atunci când aplicația trimite comenzi sau date înapoi către sistem, acestea sunt primite prin



același canal de comunicație și procesate de microcontroler.

În ceea ce privește configurarea software, microcontrolerul folosește un port serial software pentru a trimite și a primi date prin intermediul pinilor RX și TX conectați la modulul HC-05. Acest port serial virtual funcționează la aceleași principii ca și un port serial hardware, dar este gestionat prin intermediul unui program software, care permite comunicarea între microcontroler și modulul Bluetooth. În acest mod, sistemul poate trimite datele măsurate de senzorii de temperatură, puls și umiditate către telefonul mobil al utilizatorului sau poate primi comenzi de control pentru a ajusta comportamentul sistemului.

Fig 4.4 Schema electrică a senzorului de puls

În cadrul acestui proiect, senzorul MAX30100 joacă un rol crucial în măsurarea pulsului și a nivelului de oxigen din sânge (SpO₂) al pacientului. Acest senzor utilizează un protocol de comunicație I2C (Inter-Integrated Circuit), care permite un schimb eficient de date între microcontroler și senzor folosind doar două linii de semnal: SDA (Serial Data Line) și SCL (Serial Clock Line). Vom discuta despre fiecare conexiune a senzorului, rolul acestora și cum funcționează protocolul I2C pentru a transmite date între microcontroler și senzor.

Primul pin al senzorului MAX30100 este VCC, care este conectat la o sursă de alimentare de 5V. Acesta furnizează energia necesară pentru funcționarea senzorului. Deși senzorul MAX30100 poate funcționa și la o tensiune de 3.3V, conectarea la 5V garantează o performanță optimă, având în vedere că senzorul utilizează o serie de componente interne care pot necesita o tensiune mai mare pentru a funcționa corect. Al doilea pin, GND, este conectat la masa circuitului, stabilind punctul de referință de tensiune la 0V și închizând circuitul, esențial pentru buna funcționare a senzorului.

Conexiunile SDA și SCL sunt esențiale pentru comunicarea I2C între senzor și microcontroler. SDA (Serial Data Line) este folosită pentru transmiterea datelor, în timp ce SCL (Serial Clock Line) este utilizată pentru sincronizarea semnalelor de ceas între dispozitive. În cadrul acestui proiect, SDA este conectat la pinul D2 al microcontrolerului, iar SCL este conectat la pinul D1 al microcontrolerului. Aceste conexiuni permit microcontrolerului să comunice cu senzorul MAX30100 prin intermediul protocolului I2C, care este un protocol serial bidirecțional ce permite mai multor dispozitive să împartă aceleași linii de semnal, fiind astfel o metodă de comunicare economă și eficientă.

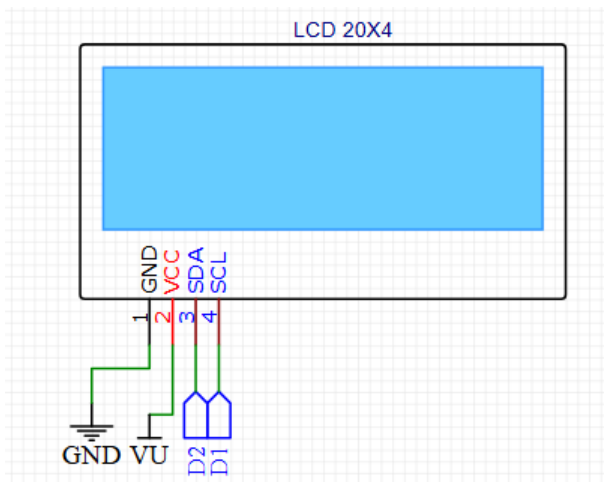
În cadrul protocolului I2C, există două componente fundamentale: master și slave. În acest caz, microcontrolerul joacă rolul de master, iar senzorul MAX30100 este slave. Comunicarea începe cu microcontrolerul care trimite un semnal de ceas pe linia SCL, urmat de un semnal de date pe linia SDA. Senzorul MAX30100, fiind un dispozitiv slave, așteaptă comenzi de la microcontrolerul master și răspunde în consecință. Datele sunt trimise într-un flux secvențial, iar la fiecare pas, senzorul MAX30100 trimite sau primește date în funcție de comenzile primite de la microcontroler.

Protocolul I2C folosește un adresaj unic pentru fiecare dispozitiv conectat pe magistrală, iar senzorul MAX30100 are adresa 0x57 în format hexazecimal. Această adresă este folosită de microcontroler pentru a identifica senzorul și pentru a iniția comunicarea cu acesta. Astfel, pentru a interacționa cu senzorul MAX30100, microcontrolerul trebuie să specifice adresa 0x57 atunci când trimite date sau primește informații. Este important de menționat că această adresă poate varia în funcție de modul în care senzorul este configurat, dar în cazul acestui proiect, adresa 0x57 este valoarea implicită.

Pașii protocolului I2C între microcontroler și senzorul MAX30100 sunt următorii: microcontrolerul trimite un semnal de ceas pe linia SCL, iar senzorul își sincronizează semnalele de date pe linia SDA. După stabilirea sincronizării, microcontrolerul adresează senzorul MAX30100 prin adresa sa I2C (0x57) și trimite un semnal de scriere sau citire, urmat de datele

necesare pentru a iniția măsurarea sau pentru a solicita măsurători de la senzor. În funcție de comanda trimisă, senzorul răspunde cu valorile corespunzătoare ale pulsului și SpO₂.

Astfel, conexiunile VCC, GND, SDA, și SCL joacă un rol fundamental în integrarea senzorului MAX30100 în sistemul de monitorizare a pacientului. Protocolul I2C asigură o comunicare eficientă între microcontroler și senzor, iar adresa I2C 0x57 permite identificarea corectă a senzorului pe magistrala I2C. Această configurație permite sistemului să obțină măsurători precise ale pulsului și oxigenării sângelui, care sunt esențiale pentru monitorizarea stării



pacientului în timp real.

Fig 4.5 Schema electrică a ecranului lcd

În cadrul acestui proiect, LCD-ul 20x4 este utilizat pentru a afișa valorile măsurate ale parametrilor vitali ai pacientului, cum ar fi pulsul, nivelul de oxigen din sânge (SpO₂) și temperatura corpului. Acesta este un afișaj cu 20 de caractere pe 4 rânduri, care poate oferi o vizualizare clară și concisă a datelor, ușor de citit de către utilizator. LCD-ul 20x4 comunică cu microcontrolerul prin protocolul I2C (Inter-Integrated Circuit), ceea ce reduce numărul de pini utilizați pe microcontroler și face integrarea mai simplă. Vom discuta despre fiecare conexiune în parte și despre pașii protocolului I2C pentru a transmite informațiile către LCD.

Primul pin al LCD-ului 20x4 este VCC, care este conectat la pinul VU al microcontrolerului. Pinul VU oferă o tensiune de 5V, care este necesară pentru alimentarea LCD-ului, deoarece acest tip de afișaj necesită o tensiune mai mare pentru a funcționa corect. VU este un pin de ieșire de 5V al microcontrolerului, care este ideal pentru a alimenta dispozitive externe ce necesită o tensiune de 5V, inclusiv LCD-ul 20x4. Al doilea pin, GND, este conectat la masa circuitului, iar această conexiune închide circuitul, stabilind punctul de referință la 0V și asigurând buna funcționare a LCD-ului.

Pentru a permite comunicarea între microcontroler și LCD-ul 20x4, se utilizează protocolul I2C, care folosește două linii principale: SDA (Serial Data Line) și SCL (Serial Clock Line). SDA

este conectat la pinul D2 al microcontrolerului, iar SCL este conectat la pinul D1 al microcontrolerului. Aceste conexiuni permit microcontrolerului să transmită date către LCD-ul 20x4 folosind protocolul I2C, care presupune un schimb de date între dispozitivele conectate pe aceeași magistrală.

Protocolul I2C este un protocol serial care permite comunicarea între mai multe dispozitive utilizând doar două linii: SDA pentru date și SCL pentru semnalul de ceas. I2C utilizează un sistem de adresare a dispozitivelor conectate, iar în cazul LCD-ului 20x4, adresa I2C este 0x27 în format hexazecimal. Această adresă este folosită de microcontroler pentru a identifica LCD-ul pe magistrala I2C și pentru a trimite date către acesta. Adresa 0x27 este valoarea implicită pentru multe dintre modulele LCD cu I2C, dar este posibil ca aceasta să fie diferită în funcție de configurarea specifică a modulelor. De obicei, această adresă poate fi citită din documentația senzorului sau ajustată prin intermediul jumperelor de pe placa de bază.

În cadrul protocolului I2C, există două roluri principale: master și slave. Microcontrolerul joacă rolul de master, iar LCD-ul 20x4 este un dispozitiv slave. Comunicarea se inițiază de către master (microcontrolerul), care trimite un semnal de ceas pe linia SCL și date pe linia SDA. În acest caz, microcontrolerul trimite instrucțiuni pentru LCD, cum ar fi comenzile de a afișa textul sau de a schimba rândurile afișajului. LCD-ul 20x4 răspunde la aceste comenzi printr-un semnal pe linia SDA.

Pașii protocolului I2C pentru a transmite date de la microcontroler la LCD sunt următorii:

1. Inițierea comunicației: Microcontrolerul trimite un semnal de ceas pe linia SCL și începe transmiterea datelor prin linia SDA. În acest moment, microcontrolerul se adresează LCD-ului folosind adresa I2C 0x27.
2. Transmiterea comenzii: Microcontrolerul trimite o comandă către LCD pentru a schimba starea afișajului sau pentru a plasa textul pe un anumit rând sau coloană al ecranului.
3. Transmiterea datelor: După comenzi, microcontrolerul poate trimite datele efective (de exemplu, caracterele care vor apărea pe ecran). Aceste date sunt transmise secvențial prin linia SDA, iar LCD-ul le afișează în timp real pe afișaj.
4. Încheierea comunicării: După ce datele au fost trimise, microcontrolerul întrerupe semnalul de ceas și oprește comunicarea.

Fig 4.6 Schema electrică a sistemului

Senzorul MAX30100 (senzorul de puls) necesită alimentare la 5V pentru a putea funcționa corect, având în vedere tehnologia internă ce implică circuite sensibile care beneficiază de un voltaj mai mare. LCD-ul 20x4 utilizează și el 5V pentru a alimenta matricea de cristale lichide, iar modulul HC-05 pentru comunicația Bluetooth, la fel, se bazează pe o alimentare de 5V pentru a putea funcționa în parametrii optimi. Astfel, toate aceste componente primesc energia necesară de la aceeași sursă de 5V, oferind o soluție simplă și eficientă pentru alimentarea întregului sistem.

Cu toate acestea, nu toate componentele din sistemul de monitorizare funcționează la 5V. De exemplu, senzorul SHT11 (care măsoară temperatura și umiditatea) și NTC-ul (senzorul de temperatură al corpului) necesită o tensiune mai mică de 3.3V pentru a funcționa corect, deoarece majoritatea senzorilor și circuitelor lor interne sunt concepute pentru a opera la această tensiune mai mică. În acest caz, microcontrolerul vine în ajutor cu un regulator intern de tensiune, care reduce tensiunea de 5V provenită din alimentarea USB la 3.3V.

Regulatorul de tensiune intern al microcontrolerului este responsabil pentru scăderea tensiunii de 5V la 3.3V, oferind o tensiune stabilă și sigură pentru senzorii SHT11 și NTC. Astfel, senzorul SHT11 și NTC-ul sunt alimentate corect și în siguranță la 3.3V, evitând orice risc de deteriorare cauzat de aplicarea unei tensiuni prea mari. Acest pas de alimentare optimă este crucial pentru asigurarea unei performanțe bune și fiabile a întregului sistem.

Prin urmare, alimentarea întregului sistem se realizează cu ajutorul unui singur cablu USB conectat la microcontroler. Tensiunea de 5V este utilizată pentru a alimenta componentele care necesită acest voltaj, iar tensiunea este redusă la 3.3V prin intermediul regulatorului de tensiune intern pentru a alimenta senzorii care necesită o tensiune mai mică. Această soluție unică de alimentare asigură funcționarea corectă a întregului sistem de monitorizare, simplificând designul și reducând necesitatea unor surse de alimentare externe suplimentare.

Pentru a înțelege mai bine consumul de energie al sistemului de monitorizare a pacientului, este important să calculăm consumul de putere al fiecărei componente și să obținem un total general. Calculul puterii consumate de fiecare componentă este esențial pentru a evalua eficiența energetică a întregului sistem, dar și pentru a estima durata de viață a bateriei sau a alimentării externe, în cazul unui sistem portabil. Vom începe prin a analiza consumul fiecărei componente și vom calcula puterea totală consumată de sistem.

În primul rând, să analizăm senzorul MAX30100, care este responsabil pentru măsurarea pulsului și a nivelului de oxigen din sânge (SpO₂). Acest senzor, având în vedere tehnologia sa bazată pe LED-uri și fotodetectori, consumă în jur de 0.1W (100mW) la funcționare normală. Este un senzor relativ eficient din punct de vedere energetic, dar având în vedere că este activ pe toată durata măsurărilor, acesta va contribui constant la consumul total de energie al sistemului.

Urmează senzorul DHT11, care măsoară temperatura și umiditatea. Acesta are un consum relativ mic, având în vedere că este un senzor de tip digital, care utilizează un procesor intern pentru a citi și transmite datele către microcontroler. Consumă aproximativ 0.02W (20mW) în timpul

măsurătorilor. Acest consum este constant, dar fiind un senzor cu un consum mic, impactul său asupra puterii totale nu este semnificativ.

Un alt senzor important este NTC-ul, care este folosit pentru a măsura temperatura corpului pacientului. Acesta face parte dintr-un divizor de tensiune, iar puterea sa de consum depinde de rezistorul utilizat în circuit și de tensiunea aplicată. Estimăm că acest senzor are un consum de aproximativ 0.03W (30mW), fiind un senzor analogic cu un consum modest. Deși nu este un senzor foarte consumator de energie, totuși joacă un rol crucial în monitorizarea temperaturii.

În ceea ce privește modulul HC-05, care este utilizat pentru comunicația Bluetooth între dispozitivul de monitorizare și aplicația Android, consumul său de energie este mai mare decât al senzorilor, deoarece funcționează continuu pentru a trimite datele la distanță. Estimăm că acest modul consumă aproximativ 0.15W (150mW) în timpul transmiterii datelor. Acesta poate consuma mai mult în perioadele de comunicare intensă, dar, în general, consumul său rămâne moderat comparativ cu alte componente ale sistemului.

LCD-ul 20x4, care afișează informațiile măsurate, este un alt consumator de energie semnificativ. Deși un LCD nu necesită o putere mare, având în vedere dimensiunea și numărul de caractere afișate, acest tip de ecran necesită aproximativ 0.2W (200mW) pentru a funcționa continuu. LCD-ul este un dispozitiv care folosește un contrast ridicat și un iluminat pe fundal pentru a asigura vizibilitatea informațiilor, iar acest lucru implică un consum de energie mai mare comparativ cu alte dispozitive de ieșire, cum ar fi OLED-urile.

În final, trebuie să luăm în considerare și consumul microcontrolerului, care este NodeMCU (bazat pe ESP8266). Deși microcontrolerul în sine este relativ eficient din punct de vedere al consumului de energie, el consumă aproximativ 0.25W (250mW) în timpul operării normale, atunci când procesează datele de la senzori și trimite informațiile prin Bluetooth. Acest consum poate varia în funcție de complexitatea sarcinilor executate, dar estimarea de 0.25W este o valoare medie adecvată.

Totalul consumului de putere al sistemului este de aproximativ 0.75W (750mW). Aceasta este puterea totală estimată pe care sistemul o va consuma în timpul funcționării normale. Acest consum nu este foarte mare, ceea ce face sistemul relativ eficient din punct de vedere energetic și potrivit pentru utilizare pe perioade mai lungi, chiar și în cazul alimentării de la o sursă portabilă, cum ar fi o baterie sau o sursă de alimentare externă. Este important să menționăm că, deși consumul de energie nu este foarte ridicat, este esențial ca sursa de alimentare să fie capabilă să furnizeze constant tensiunea necesară pentru fiecare componentă, astfel încât întregul sistem să funcționeze corect și eficient pe durata întregii perioade de utilizare.

În concluzie, consumul total estimat de energie al sistemului de monitorizare este de aproximativ 0.75W. Acesta poate fi considerat un consum relativ mic pentru un sistem electronic complex, ceea ce îl face potrivit pentru alimentare continuă pe perioade mai lungi de timp, cu o sursă de alimentare adecvată.

Capitolul 5. Dezvoltarea componentei software

Acest capitol se concentrează pe dezvoltarea componentei software a sistemului de monitorizare a pacienților, un element esențial pentru asigurarea funcționării corecte și eficiente a întregului proiect. Software-ul joacă un rol crucial, asigurând atât colectarea, prelucrarea și transmiterea datelor de la senzori, cât și afișarea și interpretarea acestora pe o interfață intuitivă pentru utilizatori.

Capitolul este structurat în două părți principale:

Prima parte descrie procesul de dezvoltare a codului Arduino pentru placa de dezvoltare **NodeMCU**, care controlează hardware-ul și gestionează comunicațiile dintre componentele sistemului. Aceasta include integrarea senzorilor, colectarea datelor, prelucrarea acestora și transmiterea lor către modulele de ieșire, precum LCD-ul și modulul Bluetooth HC-05. Tot aici, vor fi detaliate procesele logice utilizate pentru detectarea parametrilor critici și implementarea unor mecanisme de avertizare în caz de depășire a limitelor.

A doua parte se concentrează pe dezvoltarea aplicației **Android**, care reprezintă interfața utilizatorului final. Aplicația este responsabilă pentru recepționarea datelor transmise prin Bluetooth, afișarea acestora sub formă de grafice și valori numerice, precum și generarea notificărilor atunci când parametrii măsurați depășesc limitele acceptabile. Această secțiune explică arhitectura aplicației, integrarea comunicației Bluetooth, implementarea interfeței grafice și configurarea notificărilor.

5.1 Configurarea mediului de dezvoltare

Primul pas în dezvoltarea codului pentru sistemul de monitorizare a pacienților a fost configurarea mediului de dezvoltare software, mai exact instalarea și configurarea Arduino IDE. Această etapă a fost esențială pentru a putea programa și testa corect placa de dezvoltare **NodeMCU**, asigurând astfel o bază solidă pentru implementarea ulterioară a funcțiilor necesare. Alegerea Arduino IDE ca mediu de programare a fost motivată de popularitatea sa, simplitatea utilizării și suportul extins pentru o gamă largă de plăci de dezvoltare și senzori.

Procesul a început prin descărcarea și instalarea Arduino IDE de pe site-ul oficial, având grijă să selectez versiunea corespunzătoare sistemului de operare utilizat. După instalare, a fost necesară configurarea mediului pentru a recunoaște placa **NodeMCU**, care este bazată pe microcontrollerul ESP8266. În mod implicit, Arduino IDE nu include suport pentru această placă, așa că a fost necesar să adaug manual acest suport prin intermediul unui manager de plăci. În cadrul meniului „Fișier” am accesat opțiunea „Preferences” și am adăugat URL-ul specific pentru plăcile ESP8266 în câmpul destinat „Additional Board Manager URLs”. Ulterior, din meniul „Tools” -> „Board” -> „Boards Manager”, am căutat ESP8266 și am instalat pachetul corespunzător.

După ce suportul pentru NodeMCU a fost instalat, următorul pas a fost selectarea tipului de placă și a portului serial corect în Arduino IDE. Am accesat din nou meniul „Tools” și am selectat placa „NodeMCU 1.0 (ESP-12E Module)” din lista de plăci disponibile. Alegerea portului corect a fost de asemenea esențială, iar pentru aceasta am conectat placa la computer prin cablul USB și

am verificat care port serial a devenit activ în managerul de dispozitive al sistemului de operare. După identificarea portului, acesta a fost selectat în meniul „Port” din Arduino IDE pentru a permite comunicarea dintre computer și placă.

Odată ce configurația inițială a fost completă, a fost necesar să testez procesul de încărcare a codului pe placă pentru a mă asigura că totul funcționează corect. Pentru aceasta, am utilizat un exemplu simplu de cod predefinit din Arduino IDE, precum „Blink”, care face un LED să clipească la intervale regulate de timp. Am încărcat codul pe placă utilizând butonul „Upload” și am observat procesul în panoul de ieșire al IDE-ului, verificând că nu au apărut erori în timpul compilării sau al încărcării. După ce codul a fost încărcat cu succes, am observat funcționarea corectă a LED-ului de pe placă, ceea ce a confirmat că placa, conexiunile și mediul de dezvoltare erau configurate corect.

Această etapă inițială a fost esențială pentru a asigura o fundație stabilă în procesul de dezvoltare a software-ului. Configurarea corectă a mediului de dezvoltare și testarea procesului de încărcare a codului au permis identificarea eventualelor probleme de comunicare sau de compatibilitate încă din faza incipientă, reducând riscul de întârzieri ulterioare în implementarea funcționalităților mai complexe. Astfel, prin efectuarea acestor pași, am creat un mediu de lucru robust și eficient, gata să sprijine dezvoltarea completă a sistemului.

5.2 Schema logică

Schema logică prezentată ilustrează modul în care sistemul hardware și software al proiectului interacționează pentru a realiza funcționalitățile esențiale. Fiecare componentă a sistemului este inițializată, configurată și utilizată în etape bine definite, astfel încât să asigure o funcționare corectă și o comunicare eficientă între toate elementele. Scopul principal al unei astfel de scheme logice este de a reprezenta succesiunea proceselor și fluxul de date într-un mod clar și organizat, punând accent pe interacțiunile critice dintre componente.

În prima etapă a schemei logice, procesul începe cu inițializarea componentelor hardware. Această etapă este crucială, deoarece fiecare senzor și modul trebuie să fie pregătit pentru a opera corect. De exemplu, inițializarea senzorului DHT permite citirea valorilor de temperatură și umiditate, iar inițializarea LCD-ului permite afișarea datelor colectate. În plus, inițializarea modului Bluetooth este necesară pentru a permite comunicarea fără fir între sistemul hardware și aplicația Android. Totodată, modulul MAX30100 este verificat în această etapă pentru a asigura conectivitatea și funcționarea corectă, iar, în cazul unei erori, schema logică prevede afișarea unui mesaj de eroare.

După ce toate componentele sunt inițializate cu succes, schema logică intră în bucla principală a programului, unde se desfășoară toate operațiunile periodice. Această buclă asigură actualizarea constantă a datelor de la senzorul MAX30100, care măsoară ritmul cardiac și nivelul de oxigen din sânge. Tot aici, schema logică include o verificare a timpului scurs pentru a determina dacă este necesar un nou raport de date. Dacă intervalul de raportare a fost atins, se trece la procesarea datelor de la toți senzorii.

În continuare, schema logică detaliază citirea datelor de la diferitele componente. De exemplu, senzorul DHT furnizează informații despre temperatură și umiditate, iar valorile obținute sunt procesate și afișate pe ecranul LCD. În același timp, datele de la senzorul de temperatură corporală NTC sunt colectate printr-un circuit de divizor de tensiune, iar temperatura este calculată cu ajutorul formulelor specifice. Fiecare dintre aceste operațiuni este esențială pentru a oferi utilizatorului o imagine completă asupra parametrilor de sănătate monitorizați.

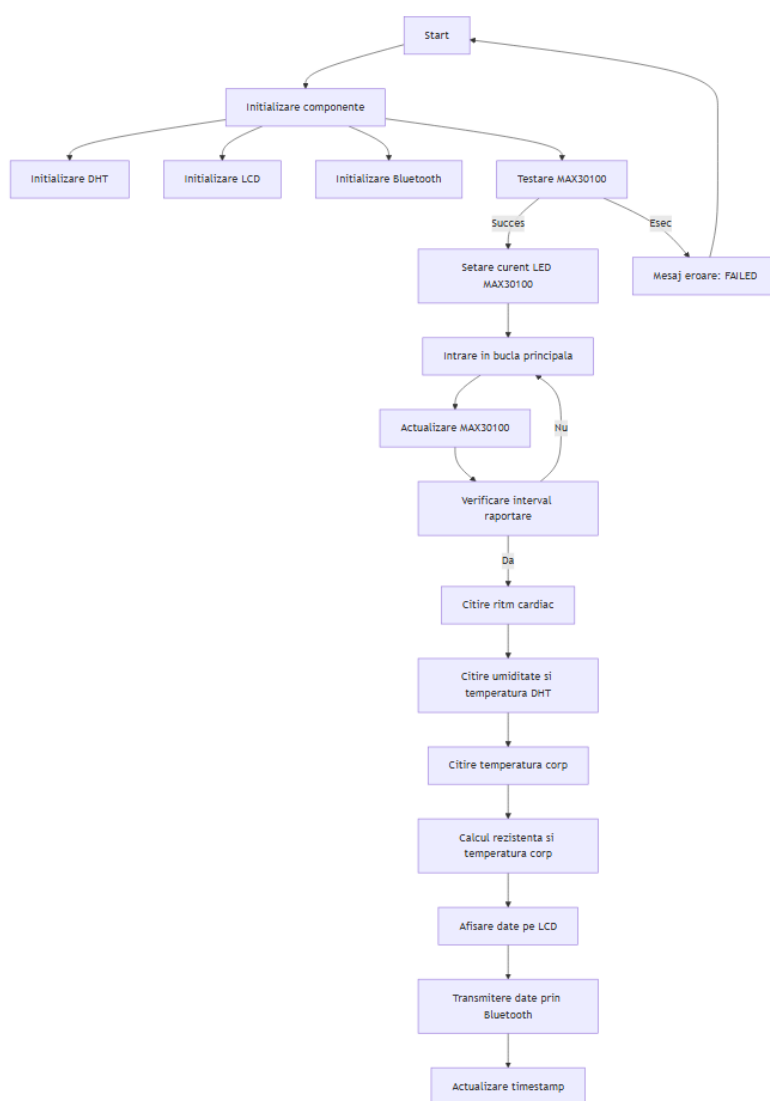


Fig 5.1 Schema logică

Un alt aspect important al schemei logice este afișarea datelor pe LCD și transmiterea acestora prin Bluetooth. LCD-ul joacă un rol critic în afișarea locală a informațiilor, astfel încât utilizatorul să poată vedea în timp real valorile ritmului cardiac, nivelului de oxigen, temperaturii corporale și condițiilor de mediu. În același timp, modulul Bluetooth trimite aceste date către aplicația Android, care le poate stoca, analiza și afișa pe grafice detaliate. Această dublă abordare

– afișare locală și transmitere la distanță – este fundamentală pentru utilitatea practică a sistemului.

La finalul fiecărei cicluri de buclă principală, schema logică prevede actualizarea timestamp-ului, astfel încât să se poată calcula corect intervalele de raportare ulterioare. Dacă intervalul de raportare nu este atins, bucla continuă să actualizeze periodic senzorii și să pregătească sistemul pentru următoarele citiri. Această structură modulară și repetitivă permite sistemului să funcționeze în mod continuu, asigurând o monitorizare constantă.

Astfel, schema logică reprezintă coloana vertebrală a întregului sistem, descriind în detaliu interacțiunile și procesele necesare pentru a colecta, prelucra și transmite datele. Fiecare etapă este interdependentă, iar funcționarea corectă a întregului sistem depinde de respectarea acestei secvențe logice bine definite.

5.3 Secvențele programului

Acest subcapitol are rolul de a detalia structura programului dezvoltat pentru implementarea funcționalităților sistemului de monitorizare. Programul a fost împărțit în mai multe secvențe distincte, fiecare având o sarcină clar definită și contribuind la realizarea obiectivelor generale ale proiectului. Această abordare modulară asigură atât o mai bună organizare a codului, cât și o flexibilitate crescută în ceea ce privește dezvoltarea și întreținerea sa.

Fiecare secvență a fost concepută astfel încât să abordeze o componentă sau un proces specific al sistemului, cum ar fi inițializarea componentelor hardware, citirea datelor de la senzori, prelucrarea valorilor, afișarea pe LCD sau transmiterea informațiilor prin modulul Bluetooth. În cadrul acestui subcapitol, fiecare parte a programului va fi explicată în detaliu, oferind o imagine clară asupra logicii și funcționării codului.

Prin această împărțire, s-a urmărit nu doar optimizarea funcționării sistemului, ci și facilitarea procesului de depanare și de adăugare a unor funcționalități noi. De asemenea, fiecare secvență a fost proiectată pentru a comunica eficient cu celelalte părți ale programului, asigurând astfel o integrare armonioasă a tuturor componentelor.

```

#include "DHT.h"
#define DHTPIN D5    // Pin digital conectat la senzorul DHT
#define DHTTYPE DHT11 // DHT 11

#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <SoftwareSerial.h>
SoftwareSerial bt(D7, D6); // RX, TX
// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 20, 4);
DHT dht(DHTPIN, DHTTYPE);

#include <Wire.h>
#include "MAX30100_PulseOximeter.h"

```

Fig 5.3 Prima parte a programului

În prima parte a codului, sunt incluse bibliotecile necesare pentru a putea interacționa cu componentele hardware ale sistemului. În primul rând, se include biblioteca „DHT.h”, care este esențială pentru a putea comunica cu senzorul DHT11. Acest senzor este utilizat pentru a măsura umiditatea și temperatura ambiantului, iar pentru a-l putea folosi corect, trebuie să definim pinul digital la care este conectat senzorul. În acest caz, senzorul DHT11 este conectat la pinul digital D5 al plăcii de dezvoltare. De asemenea, se definește tipul senzorului, care în acest caz este un DHT11, utilizat pentru a citi datele de temperatură și umiditate.

Ulterior, sunt incluse și alte biblioteci esențiale pentru funcționarea sistemului. Biblioteca „Wire.h” este folosită pentru a comunica cu dispozitivele care utilizează protocolul I2C, iar biblioteca „LiquidCrystal_I2C.h” este destinată pentru a gestiona afișajul LCD cu interfață I2C. În acest caz, LCD-ul este folosit pentru a afisa datele citite de la senzori, cum ar fi ritmul cardiac, nivelul de oxigen din sânge, temperatura și umiditatea. Pentru comunicarea cu modulul Bluetooth, care va trimite datele către aplicația Android, se include și biblioteca „SoftwareSerial.h”. Această bibliotecă este utilizată pentru a crea o comunicare serială pe pini digitali diferiți de cei utilizați pentru portul serial principal. În acest caz, modulul Bluetooth este conectat la pinii D7 (RX) și D6 (TX), iar obiectul „bt” va gestiona aceste comunicații.

Un alt pas important este inițializarea unui obiect de tip „LiquidCrystal_I2C” numit „lcd”. Acesta permite controlul LCD-ului cu interfață I2C, iar la crearea acestui obiect, i se atribuie adresa I2C a LCD-ului (0x27), dimensiunile display-ului (20 de caractere pe 4 linii) și controlul backlight-ului pentru a asigura vizibilitatea informațiilor afișate. În plus, se creează un obiect de tip „DHT” numit „dht” care este configurat pentru a comunica cu senzorul DHT11, folosind pinul D5 definit anterior. Acest obiect va fi utilizat ulterior pentru a citi valorile de temperatură și umiditate de la senzor.

În continuare, este inclusă și biblioteca „MAX30100_PulseOximeter.h”, care permite utilizarea senzorului MAX30100, ce măsoară ritmul cardiac și nivelul de oxigen din sânge prin spectrometrie de absorbție a luminii. Acest senzor utilizează protocolul I2C pentru comunicarea cu placa de dezvoltare, iar biblioteca respectivă asigură toate funcțiile necesare pentru a obține și prelucra datele de la acest senzor. Astfel, în prima parte a codului, sunt realizate toate inițializările esențiale pentru a pune în funcțiune toate componentele hardware ale sistemului, iar

acest lucru pregătește terenul pentru citirea datelor și procesarea acestora în etapele ulterioare ale programului.

Fig 5.4 A doua parte a programului

În a doua parte a codului, sunt definite o serie de constante și variabile esențiale pentru realizarea funcționării corecte a sistemului de monitorizare a semnelor vitale. Prima constantă definită este „PERIOADA_RAPORTARE_MS”, care stabilește intervalul de timp în milisecunde între

```
#define PERIOADA_RAPORTARE_MS    1000
PulseOximeter pox;
#define RT0 10000    //  $\Omega$ 
#define B 3977    // K
float a=A0;

#define VCC 5    //Supply voltage
#define R 10000    //R=10K $\Omega$ 
```

rapoartele succesive. În acest caz, valoarea este setată la 1000 ms, adică 1 secundă. Această valoare determină frecvența cu care datele sunt actualizate pe ecranul LCD și transmise prin Bluetooth. Această valoare este esențială pentru a asigura actualizarea constantă a informațiilor, astfel încât utilizatorul să poată vizualiza în timp real parametrii monitorizați.

Următorul obiect creat este un obiect de tip „PulseOximeter” numit „pox”, care este utilizat pentru a interacționa cu senzorul MAX30100, responsabil pentru măsurarea ritmului cardiac și a nivelului de oxigen din sânge. Acest obiect va fi utilizat ulterior pentru a accesa funcțiile ce permit citirea și prelucrarea datelor de la senzorul respectiv.

După aceea, sunt definite două constante, „RT0” și „B”, care sunt utilizate pentru calcularea temperaturii corpului pe baza datelor obținute de la senzorul NTC. „RT0” reprezintă rezistența la o temperatură de referință (în acest caz 10 k Ω) și este folosită într-o formulă pentru calcularea rezistenței termistorului la temperaturi diferite. Constanta „B” reprezintă un coeficient care caracterizează comportamentul senzorului NTC și este folosit în aceeași formulă pentru a obține temperatura corectă pe baza citirilor de rezistență. Aceste constante sunt esențiale pentru a obține o valoare corectă a temperaturii corpului din semnalul măsurat de senzorul NTC, care va fi folosită ulterior în calculul final.

În continuare, se definește variabila „a” și se atribuie valoarea A0, care face referire la pinul analogic de pe placa de dezvoltare. Această variabilă este utilizată pentru a prelua valoarea analogică de la senzorul NTC, care va fi ulterior procesată pentru a obține temperatura. Astfel, în a doua parte a codului, sunt pregătite toate valorile și constantele necesare pentru a obține datele relevante de la senzorii conectați și pentru a le utiliza în etapele ulterioare ale programului, asigurându-se că datele sunt calculate corect și la intervale regulate.

Fig 5.5 A treia parte a programului

```
void setup()
{
    Serial.begin(115200);
    lcd.begin();

    // Turn on the backlight and print a message.
    lcd.backlight();
    dht.begin();
    bt.begin(9600);
    if (!pox.begin()) {
```

În a treia parte a codului, este definită funcția `setup()`, care reprezintă etapa de inițializare a sistemului și de configurare a componentelor hardware. Aceasta este prima funcție care este apelată atunci când programul pornește, iar scopul acesteia este să seteze toate modulele și senzorii pentru a putea începe procesarea datelor.

În primul rând, funcția `Serial.begin(115200)` inițializează comunicarea serială cu placa de dezvoltare, folosind o viteză de 115200 bauds. Acest lucru permite trimiterea și primirea de date între placa de dezvoltare și computerul la care este conectată, fiind util pentru debug și monitorizarea valorilor citite de la senzori în timpul execuției programului. În acest mod, putem vizualiza pe monitorul serial informațiile despre starea senzorilor, ritmul cardiac, nivelul de oxigen și alte date relevante, pentru a verifica dacă sistemul funcționează corect.

Următoarea linie de cod, `lcd.begin()`, inițializează ecranul LCD, pregătind display-ul pentru a putea afișa informațiile citite de la senzori. Această funcție configurează LCD-ul pentru a putea comunica corect cu placa de dezvoltare, astfel încât datele să poată fi afișate pe ecran. La apelarea acestei funcții, se stabilește și tipul de conexiune (I2C), iar ecranul va fi pregătit să afișeze texte, valori numerice și grafice, în funcție de instrucțiunile care vor fi date mai târziu în program.

După inițializarea LCD-ului, se apelează funcția `lcd.backlight()`, care activează iluminarea de fundal a display-ului. Acesta este un pas esențial pentru asigurarea vizibilității informațiilor afișate pe LCD, mai ales în condiții de iluminare scăzută. Fără activarea iluminării de fundal, textul ar fi greu de citit, chiar și pe un ecran LCD.

În continuare, este apelată funcția `dht.begin()`, care inițializează senzorul DHT11 pentru măsurarea temperaturii și umidității. Acest pas este necesar pentru a configura corect senzorul și pentru a-l pregăti pentru a începe citirea datelor. După apelarea acestei funcții, senzorul DHT11

este gata să furnizeze date precise de temperatură și umiditate, care vor fi utilizate în cadrul sistemului de monitorizare.

În final, se apelează funcția `bt.begin(9600)`, care inițializează modulul Bluetooth HC-05 pentru a permite comunicarea wireless cu aplicația Android. Aceasta setează viteza de transfer a datelor la 9600 bauds, care este valoarea standard pentru acest tip de comunicare. Odată ce modulul Bluetooth este configurat, va fi capabil să trimită datele colectate de la senzorii DHT11, MAX30100 și NTC către aplicația mobilă, astfel încât utilizatorul să poată monitoriza în timp real semnele vitale.

Astfel, în această parte a codului, sunt inițializate toate componentele hardware ale sistemului și se configurează setările necesare pentru ca acestea să funcționeze corect. Fiecare componentă este pregătită pentru a începe să interacționeze cu restul sistemului și să furnizeze datele necesare

```
    if (!pox.begin()) {
    Serial.println("FAILED");
    for(;;);
    } else {
    Serial.println("SUCCESS");
    }
    pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);

    pinMode(A0, INPUT);
    T0 = 25 + 273.15;
}
```

monitorizării parametrilor de sănătate.

Fig 5.6 A patra parte a programului

În a patra parte a codului sunt realizate câteva verificări esențiale pentru a se asigura că senzorul MAX30100 funcționează corect și că sistemul este pregătit să preia și să proceseze datele de la acest senzor.

Prima secțiune a codului verifică dacă senzorul MAX30100 este corect inițializat. Aceasta se face prin apelarea metodei `pox.begin()`. Dacă senzorul nu a fost inițializat corect (de exemplu, dacă nu există nicio conexiune validă între senzor și placa de dezvoltare), condiția `if (!pox.begin())` va fi adevărată, iar programul va intra în ramura de cod corespunzătoare. În acest caz, se va afișa mesajul "FAILED" pe monitorul serial, indicând faptul că inițializarea senzorului a eșuat. După aceea, programul va intra într-un ciclu infinit cu `for(;;)`, ceea ce înseamnă că execuția va fi oprită și nu va mai continua niciun alt proces. Acest comportament este util pentru a preveni funcționarea unui sistem cu un senzor nefuncțional, deoarece datele incorecte sau inexistente ar putea duce la erori în monitorizarea semnelor vitale.

Pe de altă parte, dacă senzorul MAX30100 este inițializat cu succes, condiția else va fi executată. În acest caz, mesajul "SUCCESS" va fi afișat pe monitorul serial, indicând faptul că senzorul este gata să furnizeze date corecte. Este o verificare utilă pentru a confirma că sistemul funcționează așa cum este de așteptat și că senzorul MAX30100 a fost detectat corespunzător de către placa de dezvoltare.

După inițializarea senzorului, se apelează funcția `pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA)`, care setează curentul pentru LED-ul infraroșu al senzorului MAX30100. Această setare este importantă pentru a asigura o măsurare corectă a pulsului și a nivelului de oxigen din sânge. Curentul LED-ului poate influența precizia citirii, iar valoarea aleasă în acest caz, `MAX30100_LED_CURR_7_6MA`, reprezintă o valoare standard de curent pentru acest senzor, care asigură o bună performanță.

La finalul acestei secțiuni de cod, se configurează pinul A0 al plăcii de dezvoltare ca intrare, folosind `pinMode(A0, INPUT)`. Acest pin va fi utilizat pentru citirea valorilor de la senzorul NTC, care măsoară temperatura corpului. Prin setarea pinului ca intrare, se permite citirea semnalului analogic de la senzor și procesarea acestuia mai târziu în program.

În final, se calculează valoarea variabilei T_0 care reprezintă temperatura de referință pentru senzorul NTC, adăugând 25 grade Celsius (temperatura ambientală standard) la valoarea 273.15 pentru a obține valoarea în Kelvin. Astfel, T_0 va reprezenta temperatura de referință în unități de Kelvin, care va fi utilizată ulterior pentru calculul temperaturii corpului pe baza rezistenței senzorului NTC.

Astfel, această parte a codului este responsabilă pentru inițializarea senzorului MAX30100 și pentru configurarea altor parametri necesari pentru măsurarea corectă a semnalelor vitale. Codul asigură, de asemenea, că dacă senzorul nu este disponibil sau nu funcționează corect, sistemul nu

```
void loop()
{
    pox.update();

    if (millis() - tsUltimRaport > PERIOADA_RAPORTARE_MS) {
        int ritmCardiac = pox.getHeartRate();
```

va continua să ruleze cu date eronate.

Fig 5.7 A cincea parte a programului

În a cincea parte a codului, începe bucla principală a programului, care se va repeta continuu atâta timp cât sistemul este activ. Este locul unde se realizează actualizarea și prelucrarea datelor de la senzorii conectați la sistem.

Prima linie din această secțiune, `pox.update()`, este utilizată pentru a actualiza valorile de măsurare ale senzorului MAX30100, care monitorizează pulsul și nivelul de oxigen din sânge (SpO2). Această comandă este esențială pentru a obține cele mai recente citiri de la senzor și pentru a le prelucra ulterior. Senzorul MAX30100 funcționează pe principiul măsurării absorbției de lumină de către sânge, iar această funcție garantează că datele colectate sunt actualizate și corecte înainte de a fi folosite în program.

Următorul bloc de cod din această secțiune este condiția `if (millis() - tsUltimRaport > PERIOADA_RAPORTARE_MS)`. Aici se face verificarea pentru a decide când să actualizeze raportul cu noile valori citite. Funcția `millis()` returnează timpul în milisecunde care a trecut de la pornirea programului, iar variabila `tsUltimRaport` conține timpul ultimei raportări. Prin urmare, această condiție verifică dacă a trecut un interval de timp mai mare decât perioada stabilită pentru raportare (specificată de constantă `PERIOADA_RAPORTARE_MS`, care este setată la 1000 de milisecunde sau 1 secundă). Dacă această condiție este adevărată, înseamnă că este timpul să actualizezi valorile și să raportezi noile măsurători.

Dacă condiția este îndeplinită, programul va executa instrucțiunile din interiorul blocului `if`, care se referă la obținerea datelor de la senzorul MAX30100 și la prelucrarea acestora. În acest caz, se obține ritmul cardiac folosind metoda `pox.getHeartRate()`, care returnează valoarea pulsului în bătăi pe minut (bpm). Această valoare este stocată în variabila `ritmCardiac` și va fi utilizată ulterior pentru a afișa pulsul pe LCD și pentru a trimite datele către aplicația Android prin modulul Bluetooth HC-05.

În continuare, vor fi preluate valorile de umiditate și temperatură de la senzorul DHT11. Acestea sunt obținute prin apelul funcțiilor `dht.readHumidity()` și `dht.readTemperature()`, care returnează valoarea umidității relative și a temperaturii ambiante în grade Celsius. Aceste valori sunt importante pentru monitorizarea mediului în care se află pacientul, dar și pentru a detecta eventuale condiții care ar putea afecta sănătatea acestuia, precum o temperatură prea ridicată sau o umiditate prea scăzută.

După obținerea valorilor de la senzori, sistemul va calcula și va actualiza afisajul pe ecranul LCD. În cadrul acestei secțiuni a codului, se setează cursorul LCD-ului și se afișează informațiile relevante, cum ar fi ritmul cardiac (BPM), nivelul de oxigen din sânge (SpO2), temperatura ambiantă și umiditatea, dar și temperatura corpului calculată pe baza senzorului NTC. Aceste informații sunt afișate într-un format ușor de citit, fiecare dată fiind actualizată la fiecare secundă, conform perioadei de raportare stabilite anterior.

De asemenea, datele sunt trimise prin conexiunea Bluetooth către aplicația Android, folosind modulul HC-05. Aceste informații sunt trimise în formatul corespunzător (valorile ritmului cardiac, SpO2, temperatura ambiantă, temperatura corpului și umiditatea), separate de punct și virgulă, pentru a fi prelucrate corect de aplicația mobilă.

În finalul acestei secțiuni, variabila `tsUltimRaport` este actualizată cu valoarea curentă a timpului, obținută prin `millis()`. Acest lucru permite ca, la următoarea trecere prin bucla principală, să se verifice din nou dacă a trecut perioada de raportare stabilită și să se actualizeze valorile pentru o nouă raportare.

Această parte a codului este esențială pentru buna funcționare a sistemului, deoarece gestionează actualizarea continuă a datelor și comunicarea acestora atât către utilizatorul local (prin ecranul LCD), cât și către aplicația mobilă (prin Bluetooth).

```
float umiditate = dht.readHumidity();  
float temperatura = dht.readTemperature();  
Serial.print("Ritm cardiac: ");  
Serial.print(ritmCardiac);  
Serial.print(" bpm / SpO2: ");
```

Fig 5.8 A șasea parte a programului

În a șasea parte a codului, sunt colectate și procesate datele de la senzorii de mediu, în special umiditatea și temperatura, folosind senzorul DHT11. Mai exact, variabilele umiditate și temperatura sunt actualizate cu valorile citite de la senzor. Apelul funcției `dht.readHumidity()` returnează valoarea umidității relative din aer, în timp ce funcția `dht.readTemperature()` returnează temperatura ambientală, măsurată în grade Celsius. Aceste două date sunt esențiale pentru monitorizarea mediului în care se află pacientul, deoarece schimbările bruște ale temperaturii sau umidității pot influența starea de sănătate.

În continuare, informațiile despre ritmul cardiac și nivelul de oxigen din sânge (SpO2), care au fost deja preluate și procesate de la senzorul MAX30100 în secțiunile anterioare ale codului, sunt afișate pe monitorul serial pentru a putea fi vizualizate în timpul testelor. Acest lucru este realizat cu ajutorul funcției `Serial.print()`, care trimite datele către portul serial și le face vizibile în IDE-ul Arduino. Primul mesaj transmis este „Ritm cardiac:”, urmat de valoarea ritmului cardiac, care este stocată în variabila `ritmCardiac`, în bătăi pe minut (bpm). Aceasta este urmată de un mesaj care indică valoarea SpO2, calculată de senzorul MAX30100. Aceste informații sunt extrem de utile pentru monitorizarea stării pacientului și pot ajuta la detectarea unor eventuale probleme de sănătate.

Prin urmare, această secțiune de cod are rolul de a actualiza variabilele cu datele preluate de la senzorii DHT11, iar ulterior, să le afișeze pe monitorul serial pentru a verifica dacă citirile se efectuează corect. În această etapă, codul permite utilizatorului să vizualizeze în timp real atât ritmul cardiac și nivelul SpO2, cât și valorile de temperatură și umiditate, pentru a asigura o monitorizare continuă și exactă a stării de sănătate a pacientului.

Este important de menționat că aceste date sunt folosite și pentru actualizarea ecranului LCD, unde sunt afișate într-un format clar și ușor de citit, însă în această secțiune a codului, informațiile sunt pur și simplu trimise către portul serial pentru verificare și debugging. Acesta este un pas important în procesul de dezvoltare al sistemului, deoarece permite testarea funcționării corecte a senzorilor și afișarea lor într-un mediu controlat.

```

lcd.setCursor(0,0);
lcd.print("Puls:");
lcd.print(ritmCardiac);
lcd.print("BPM");
lcd.setCursor(0,1);
lcd.print("Spo2:");
lcd.print(pox.getSpO2());
lcd.setCursor(7,1);
lcd.print("%");

```

Fig 5.9 A șaptea parte a programului

În a șaptea parte a codului, se realizează afișarea valorilor obținute de la senzorul de ritm cardiac și de la senzorul de oxigen din sânge (SpO2) pe ecranul LCD. Folosind funcțiile de control ale bibliotecii LiquidCrystal_I2C, programul setează locațiile specifice pe ecran pentru a vizualiza aceste date într-un mod clar și organizat.

În prima linie de cod, `lcd.setCursor(0,0);` stabilește poziția cursorului pe colțul din stânga sus al ecranului LCD, adică la poziția (0,0). Apoi, funcția `lcd.print("Puls:");` afișează cuvântul „Puls:” pe ecran, pentru a semnaliza utilizatorului că valoarea care urmează să fie afișată reprezintă ritmul cardiac. Imediat după aceea, ritmul cardiac, care este stocat în variabila `ritmCardiac`, este afișat cu ajutorul funcției `lcd.print(ritmCardiac);`, urmat de unitatea de măsură „BPM” (bătăi pe minut), care este concatenată la valoarea numerică afișată. Acest lucru permite vizualizarea ritmului cardiac al pacientului în timp real.

Următorul pas în acest bloc de cod constă în setarea cursorului pe linia 2, prima coloană, folosind comanda `lcd.setCursor(0,1);`. Aceasta poziționează cursorul pe a doua linie a ecranului LCD. Apoi, este afișat cuvântul „Spo2:”, care semnalează utilizatorului că următoarea valoare va reprezenta nivelul de oxigen din sânge, sau SpO2. Pentru a afișa valoarea SpO2, se folosește funcția `lcd.print(pox.getSpO2());`, care preia valoarea din senzorul MAX30100 și o afișează pe ecran. La sfârșit, este adăugat simbolul „%”, pentru a indica faptul că valoarea reprezintă procentajul de oxigen din sânge.

Această secțiune de cod are rolul de a vizualiza în timp real două dintre cele mai importante date monitorizate de sistem: ritmul cardiac și nivelul de oxigen din sânge. Aceste informații sunt esențiale pentru evaluarea stării de sănătate a pacientului, iar utilizarea unui ecran LCD pentru a le afișa permite un acces rapid și clar la aceste date, fără a fi necesar un alt dispozitiv pentru a le vizualiza. Prin urmare, această secțiune ajută la monitorizarea continuă a pacientului, oferind informații vizuale și ușor de înțeles pe un ecran accesibil.

Prin combinarea acestora cu celelalte informații afișate pe ecran, cum ar fi temperatura și umiditatea, se creează o interfață completă și intuitivă pentru utilizator, permițând un control și o evaluare rapidă a stării de sănătate a pacientului.

```
Read = analogRead(A0);           //Acquisition analog value Read
Read = (5.00 / 1023.00) * Read;  //Conversion to voltage
VR = VCC - Read;
RT = Read / (VR / R);           //Resistance of RT

ln = log(RT / RT0);
Temp = (1 / ((ln / B) + (1 / T0))); //Temperature from sensor

Temp = Temp - 273.15;           //Conversion to Celsius
```

Fig 5.10 A opta parte a programului

În a opta parte a codului, se realizează măsurarea temperaturii corpului prin intermediul unui senzor de temperatură, bazat pe un senzor NTC. Procesul începe cu citirea valorii analogice de pe pinul A0 al plăcii de dezvoltare, prin intermediul funcției `analogRead(A0)`. Această funcție returnează o valoare între 0 și 1023, care reprezintă o măsurătoare digitalizată a tensiunii de intrare pe pinul A0, corespunzătoare unui semnal analogic continuu.

După obținerea acestei valori, programul o convertește într-o tensiune reală folosind formula: $\text{Read} = (5.00 / 1023.00) * \text{Read}$. Aceasta presupune că placa de dezvoltare are o tensiune de alimentare de 5V, iar valoarea citită de la senzor este împărțită la valoarea maximă a ADC-ului (1023) pentru a obține o tensiune reală între 0 și 5V.

Următoarea etapă este calculul valorii rezistenței senzorului de temperatură, denumită R_T , folosind formula: $R_T = \text{Read} / (V_R / R)$. Aici, V_R este tensiunea măsurată la terminalul senzorului, iar R reprezintă o valoare de referință a rezistorului din circuit, în acest caz 10k Ω . Această formulă derivă din principiul unui divizor de tensiune, unde senzorul de temperatură face parte dintr-un circuit care convertește temperatura într-o variație de rezistență, iar aceasta este măsurată sub forma unui semnal electric.

Odată ce valoarea rezistenței R_T este obținută, programul calculează temperatura utilizând formula matematică bazată pe legea lui Steinhart-Hart, care este folosită pentru senzori de temperatură NTC (Negative Temperature Coefficient). Acesta se realizează prin calcularea logaritmului natural al raportului între rezistența senzorului și valoarea de referință a rezistenței la temperatura standard, R_{T0} . Formula utilizată pentru a obține logaritmul este $\ln = \log(R_T / R_{T0})$, unde R_{T0} reprezintă rezistența la o temperatură de referință de 25°C (adică 10000 Ω).

După obținerea valorii logaritmului, se poate calcula temperatura absolută în Kelvin utilizând formula termodinamică: $\text{Temp} = (1 / ((\ln / B) + (1 / T_0)))$. Aici, B este o constantă specifică senzorului de temperatură, iar T_0 reprezintă temperatura de referință în Kelvin (298.15K, corespunzător lui 25°C). Rezultatul este temperatura în grade Kelvin, iar ultima etapă este conversia acesteia în grade Celsius, folosind formula: $\text{Temp} = \text{Temp} - 273.15$.

Prin această secțiune de cod, temperatura corpului pacientului este calculată precis, având în vedere caracteristicile senzorului NTC, care variază rezistența în funcție de temperatură. Aceste măsurători sunt esențiale pentru monitorizarea stării de sănătate a pacientului, iar codul asigură că datele de temperatură sunt obținute corect și utilizate ulterior pentru a oferi informații relevante asupra stării acestuia. Calculul valorii de temperatură în grade Celsius este un pas esențial pentru a putea interpreta corect datele și pentru a le prezenta în mod clar utilizatorului sau pe ecranul LCD.

```
lcd.setCursor(0,2);  
lcd.print("T:");  
lcd.print(temperatura);  
lcd.print("C ");  
lcd.print("H:");  
lcd.print(umiditate);  
lcd.print("%");  
lcd.setCursor(0,3);  
lcd.print("TempCorp:");  
lcd.print(Temp);  
lcd.print("C");
```

Fig 5.11 A noua parte a programului

În a noua parte a codului, se realizează actualizarea ecranului LCD cu valorile măsurate și procesate de senzorii sistemului. Mai exact, se afișează informațiile referitoare la temperatura ambientală, umiditatea și temperatura corpului.

Primul pas este setarea poziției cursorului pe ecranul LCD. Acest lucru se face folosind comanda `lcd.setCursor(0,2);`, care plasează cursorul la începutul celei de-a treia linii a ecranului LCD (linia 2, deoarece număratoarea începe de la 0). După ce cursorul este plasat, se afișează valoarea temperaturii măsurate de senzorul DHT11, cu ajutorul comenzii `lcd.print("T:");`, care scrie caracterul "T:" pe prima coloană a liniei respective, urmat de valoarea temperaturii. Aceasta este obținută din variabila `temperatura`, care a fost citită anterior de la senzorul DHT11. Temperatura este exprimată în grade Celsius și este afișată după textul "T:". După această valoare, se adaugă unitatea de măsură "C", indicând faptul că măsurătoarea se referă la temperatura ambientală.

Pe lângă temperatura ambientală, se mai afișează și umiditatea, care este citită tot de la senzorul DHT11. Poziția cursorului este păstrată pe aceeași linie și se folosește din nou comanda `lcd.print("H:");` pentru a adăuga textul "H:" înaintea valorii umidității. Aceasta este stocată în variabila `umiditate` și este afișată direct pe ecran, urmată de simbolul "%" pentru a indica unitatea de măsură a umidității relative.

După ce informațiile legate de temperatura ambientală și umiditatea relativă sunt afișate pe prima linie a ecranului LCD, cursorul se mută pe linia a patra a LCD-ului cu comanda `lcd.setCursor(0,3);`. Aici, se afișează temperatura corpului pacientului, care a fost calculată anterior folosind senzorul NTC. La fel ca și în cazul temperaturii ambientale, se afișează textul "TempCorp:", urmat de valoarea procesată a temperaturii corpului, stocată în variabila `Temp`.

Aceasta reprezintă temperatura internă a corpului pacientului și este afișată în grade Celsius, urmată de simbolul "C". Astfel, pe ecranul LCD sunt vizualizate simultan toate informațiile esențiale pentru monitorizarea stării de sănătate a pacientului: temperatura ambientală, umiditatea relativă și temperatura corpului.

Această secțiune a codului este esențială pentru interfața cu utilizatorul, oferind o modalitate rapidă și clară de a vizualiza datele colectate de la senzori. Utilizatorul poate verifica în timp real temperatura ambientală, umiditatea și temperatura corpului pacientului, toate afisate pe ecranul LCD, fără a fi nevoie de un dispozitiv suplimentar pentru a vizualiza aceste informații.

```
bt.print(ritmCardiac);  
bt.print(";");  
bt.print(pox.getSpO2());  
bt.print(";");  
bt.print(Temp);  
bt.print(";");  
bt.print(temperatura);  
bt.print(";");  
bt.print(umiditate);  
bt.println(";");  
Serial.print(pox.getSpO2());  
Serial.println("%");  
tsUltimRaport = millis();  
}
```

Fig 5.12 A zecea parte a programului

În a zecea parte a codului, datele colectate de la senzori sunt trimise către aplicația Android prin intermediul modului Bluetooth HC-05, folosind protocolul serial. Această secțiune este crucială pentru a transmite valorile măsurate într-un mod accesibil și ușor de procesat pe dispozitivul mobil, permițând astfel utilizatorului să vizualizeze datele într-un format adecvat pe aplicația dedicată.

Primul pas este trimiterea valorii ritmului cardiac (stocată în variabila `ritmCardiac`) către aplicația Android. Acest lucru se realizează folosind comanda `bt.print(ritmCardiac);`, care trimite valoarea respectivă prin portul serial virtual configurat pe pinurile D6 (RX) și D7 (TX) ale plăcii de dezvoltare. După transmiterea ritmului cardiac, sunt adăugate separatorul ";" pentru a delimita datele trimise și pentru a permite aplicației Android să le interpreteze corect.

Următorul pas este trimiterea valorii SpO2 (saturația oxigenului din sânge) citită de la senzorul MAX30100, folosind comanda `bt.print(pox.getSpO2());`. La fel ca în cazul ritmului cardiac, această valoare este urmată de separatorul ";", pentru a asigura că datele sunt corect delimitate.

După aceea, temperatura corpului pacientului (stocată în variabila `Temp`) este trimisă și ea, folosind comanda `bt.print(Temp);`, urmată de separatorul ";". Acesta este urmat de trimiterea valorii temperaturii ambientale citită de la senzorul DHT11, stocată în variabila `temperatura`, prin

comanda `bt.print(temperatura);`. Această valoare, de asemenea, este urmată de separatorul `";"` pentru a facilita procesarea ulterioară a datelor.

În continuare, valoarea umidității relative (stocată în variabila `umiditate`) este trimisă prin comanda `bt.print(umiditate);`, urmată de separatorul `";"`. La finalul secvenței de date, se adaugă comanda `bt.println(";");`, care trimite un caracter de linie nouă, semnalând finalul unui pachet de date și pregătirea pentru următoarele date ce vor fi trimise.

Pe lângă transmiterea datelor către aplicația mobilă, codul trimite și aceleași informații către monitorul serial conectat la placa de dezvoltare, folosind comanda `Serial.print(pox.getSpO2());` pentru a trimite valoarea `SpO2` și `Serial.println("%");` pentru a afișa semnul procentual la sfârșitul valorii. Acesta este un instrument util pentru monitorizarea directă a datelor pe computer, permițând verificarea valorilor măsurate în timp real.

În final, este actualizat timestamp-ul ultimei rapoarte de date folosind comanda `tsUltimRaport = millis();`, care înregistrează timpul curent în milisecunde de la pornirea programului. Această valoare este utilizată pentru a determina intervalul de timp dintre două rapoarte succesive și pentru a asigura că datele sunt trimise la intervalele corecte, conform valorii setate în constantă `PERIOADA_RAPORTARE_MS`.

Această secțiune a codului este esențială pentru interacțiunea dintre sistemul hardware și aplicația Android, deoarece asigură transmisia eficientă a datelor către dispozitivul mobil, pentru ca utilizatorul să poată monitoriza în timp real parametrii fiziologici ai pacientului.

5.4 Dezvoltarea aplicației Android

MIT App Inventor este o platformă de dezvoltare vizuală care permite utilizatorilor să creeze aplicații mobile pentru dispozitive Android într-un mod simplu și intuitiv, fără a necesita cunoștințe avansate de programare. Platforma a fost creată inițial de Universitatea din Massachusetts (MIT) pentru a ajuta utilizatorii, inclusiv pe cei fără experiență de programare, să dezvolte aplicații mobile folosind o interfață grafică bazată pe blocuri de cod, într-un stil similar cu programarea vizuală. MIT App Inventor a fost lansată în 2010 și este utilizată în mod frecvent în educație și în diverse proiecte inovatoare, datorită accesibilității sale și a ușurinței de utilizare. Această platformă a evoluat semnificativ în ultimii ani, adăugând suport pentru funcționalități avansate, inclusiv integrarea cu hardware-ul extern, API-uri web și baze de date online, extinzându-și astfel capabilitățile.

Platforma MIT App Inventor se bazează pe o interfață de tip drag-and-drop (trage și plasează), unde utilizatorii pot construi aplicații folosind blocuri de cod predefinite care sunt conectate între ele. Aceste blocuri reprezintă diferite funcționalități, iar utilizatorii pot construi aplicații pentru dispozitive mobile fără a fi nevoie să scrie cod tradițional. App Inventor folosește un limbaj de programare bazat pe blocuri, ce poate fi înțeles rapid și ușor de toți utilizatorii, indiferent de nivelul lor de cunoștințe tehnice. Acesta poate fi folosit pentru a crea aplicații care includ interfețe de utilizator, gestionarea datelor, conectivitate cu dispozitive externe și multe altele.

Unul dintre punctele forte ale MIT App Inventor este capacitatea sa de a interacționa cu hardware-ul extern, cum ar fi senzorii și modulele Bluetooth, ceea ce îl face o alegere ideală pentru proiectele care implică Internetul Lucrurilor (IoT). Platforma suportă diverse protocoale de comunicație, inclusiv Bluetooth, Wi-Fi și comunicații prin internet, permițând dezvoltarea de aplicații mobile care pot comunica cu dispozitivele hardware externe. În acest context, App Inventor este extrem de util în proiectele care implică microcontrolere, cum ar fi Arduino, ESP8266 sau ESP32, oferind un mod ușor și rapid de a crea aplicații care pot comunica cu aceste dispozitive.

În cadrul proiectului nostru, MIT App Inventor a fost folosit pentru a dezvolta aplicația mobilă care interacționează cu sistemul hardware de monitorizare a parametrilor vitali. Aplicația este destinată să primească datele trimise de dispozitivele hardware prin Bluetooth, să le interpreteze și să le afișeze într-o formă ușor de înțeles pentru utilizator. Aceasta permite utilizatorilor să monitorizeze ritmul cardiac, nivelul de saturație a oxigenului (SpO2), temperatura ambientală și temperatura corpului. Aplicația folosește modulul Bluetooth HC-05 pentru a primi datele transmise de la dispozitivul hardware și pentru a le prezenta într-o interfață prietenoasă, incluzând valori numerice, grafice și diagrame.

Un alt avantaj important al MIT App Inventor este că permite integrarea ușoară cu bazele de date online, ceea ce poate fi util pentru stocarea și gestionarea datelor colectate pe termen lung. De asemenea, aplicația mobilă poate trimite datele către un server cloud pentru a fi stocate și analizate, oferind o soluție de monitorizare de la distanță. Platforma App Inventor oferă și funcționalități de interacțiune cu diverse servicii externe, cum ar fi API-urile pentru web sau integrarea cu servicii de mesagerie, care pot fi utile pentru trimiterea alertelor sau notificărilor în cazul în care valorile măsurate depășesc anumite limite predefinite.

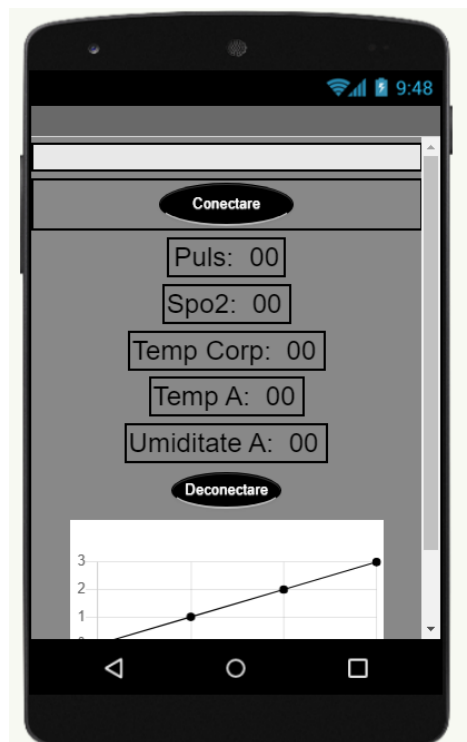
Istoricul platformei MIT App Inventor arată un progres constant în direcția unui acces mai larg și mai simplu la dezvoltarea de aplicații mobile. De la începuturi, când MIT App Inventor era utilizat în principal în scopuri educaționale, până la versiunea actuală, care permite dezvoltarea de aplicații complexe și funcționale, platforma a evoluat pentru a satisface cerințele tot mai mari ale utilizatorilor din diverse domenii. În prezent, MIT App Inventor este folosit pe scară largă în educație, în special pentru predarea conceptelor de programare și pentru proiectele de IoT, dar și de dezvoltatori care doresc să creeze rapid prototipuri de aplicații.

Capabilitățile platformei includ nu doar programarea vizuală, dar și integrarea cu hardware-ul, accesul la internet și interacțiunea cu servicii externe. Aceste funcționalități fac din MIT App Inventor un instrument extrem de puternic și versatil, care poate fi utilizat în multe domenii de aplicații. Cu toate acestea, este important de menționat că platforma are și anumite limitări, în special în ceea ce privește complexitatea aplicațiilor, comparativ cu platformele de dezvoltare profesională. Cu toate acestea, MIT App Inventor rămâne o alegere excelentă pentru prototipurile rapide și pentru utilizatorii care doresc să dezvolte aplicații mobile cu un efort minim și fără a fi necesar să aibă cunoștințe de programare avansate.

În concluzie, MIT App Inventor a fost o alegere perfectă pentru dezvoltarea aplicației mobile a proiectului nostru, datorită interfeței sale simple, capacităților sale de integrare cu hardware-ul extern și sprijinului pentru dezvoltarea rapidă de aplicații. Această platformă a permis crearea

unei aplicații funcționale, care colectează și afișează datele în timp real, contribuind la succesul proiectului.

Interfața utilizatorului a aplicației reprezintă un element fundamental al designului său, fiind esențială pentru interacțiunea utilizatorului cu aplicația și pentru vizualizarea datelor importante. Această interfață este destinată unui dispozitiv mobil, precum un telefon sau o tabletă, și este concepută pentru a oferi o metodă intuitivă și ușor de utilizat de monitorizare a semnelor vitale ale unei persoane, fiind, astfel, specifică aplicațiilor medicale. Aceasta poate include măsurători ale pulsului, saturației de oxigen în sânge (SpO2), temperaturii corporale și umidității



ambientale, printre altele.

Fig 5.13 Intefața utilizatorului

Unul dintre elementele centrale ale interfeței este titlul aplicației, care se află în partea superioară a ecranului și se numește „Display hidden components in Viewer”. Aceasta sugerează că utilizatorii ar putea avea opțiunea de a activa sau dezactiva vizualizarea anumitor componente ale aplicației, facilitând astfel personalizarea interfeței conform nevoilor și preferințelor utilizatorului.

În partea dreaptă sus a ecranului, este afișată ora curentă (de exemplu, 9:48), o funcționalitate standard care oferă utilizatorilor un indicator temporal ușor accesibil în orice moment al utilizării aplicației. Ora afișată poate fi utilă în contextul monitorizării continue a semnelor vitale, permițând utilizatorilor să coreleze datele obținute cu timpul real al măsurărilor.

De asemenea, în interfață sunt prezentate două butoane esențiale pentru controlul dispozitivului: Conectare/Deconectare. Aceste butoane sunt utilizate pentru a stabili și a întrerupe conexiunea aplicației cu senzorii sau cu un dispozitiv extern, posibil pentru transmiterea și actualizarea datelor medicale. Această funcționalitate este esențială în contextul aplicațiilor care depind de date provenite din dispozitive externe, cum ar fi monitoarele de puls sau senzori de temperatură.

În ceea ce privește datele fiziologice colectate de aplicație, interfața include patru câmpuri vizibile pentru valori: Puls, SpO2, Temp Corp și Temp A, fiecare cu o valoare implicită de „00”. Aceste câmpuri sunt destinate să afișeze informațiile relevante pentru sănătatea utilizatorului: pulsul (bătăile pe minut), saturația de oxigen în sânge (SpO2), temperatura corporală și temperatura ambientală. În acest moment, valorile afișate sunt „00”, ceea ce sugerează că senzorii nu sunt conectați sau nu au fost încă obținute măsurători. Aceasta este o stare inițială comună pentru aplicațiile de monitorizare a semnelor vitale, înainte ca dispozitivele de măsurare să fie activate.

Mai mult, interfața include și un câmp pentru Umiditate A, care afișează valoarea umidității mediului înconjurător, o altă măsurătoare relevantă în contextul monitorizării condițiilor de sănătate. La fel ca celelalte câmpuri, valoarea este setată la „00”, indicând lipsa datelor sau nefuncționarea senzorilor.

Un alt element important al interfeței este graficul liniar inclus în aplicație. Acesta ar putea reprezenta evoluția în timp a unuia dintre parametrii monitorizați, dar fiind gol în acest moment, nu poate fi utilizat pentru a furniza informații utile. Graficul liniar este un instrument vizual valoros, care permite utilizatorilor să urmărească progresul sau variațiile semnelor vitale pe o perioadă mai lungă de timp, făcând analiza mai ușor de înțeles și de urmărit.

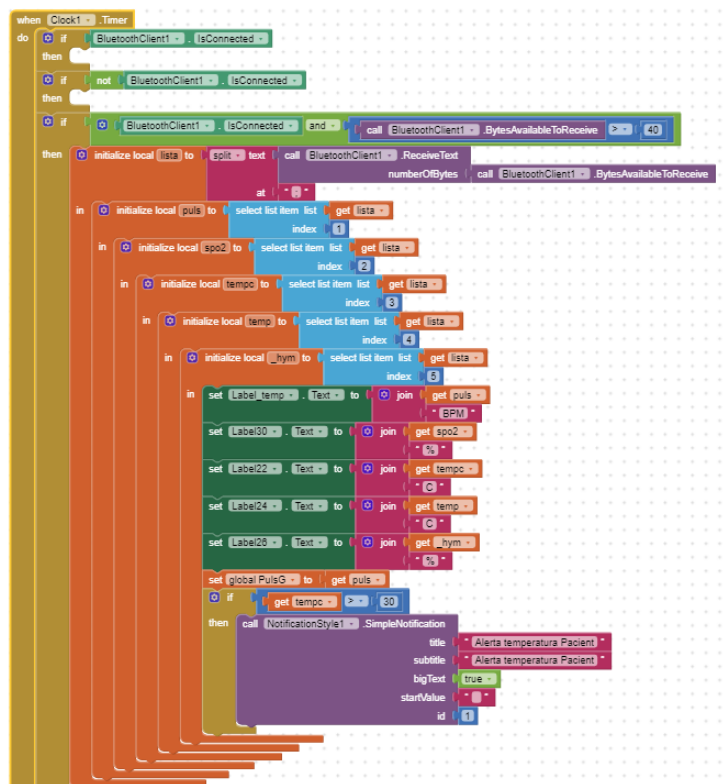
În ansamblu, interfața de utilizator a aplicației este proiectată să fie simplă și eficientă, oferind utilizatorilor o experiență de monitorizare clară și directă a semnelor vitale. Cu toate acestea, din cauza valorilor inițiale „00” și a absenței etichetelor pentru axele graficului, interfața nu poate fi pe deplin evaluată din punct de vedere funcțional fără o conexiune activă la senzori sau fără o utilizare efectivă a aplicației. Pe măsură ce senzorii sunt conectați și datele sunt colectate, aplicația va deveni mult mai utilă și interactivă, permițând utilizatorilor să obțină informații esențiale despre starea lor de sănătate într-un mod clar și accesibil.

În prima parte de cod din aplicația MIT Inventor, se gestionează procesul de conectare și primire a datelor prin Bluetooth de la dispozitivul microcontrollerului sau senzorii conectați. Codul începe cu inițierea unui cronometru periodic, denumit Clock1.Timer, care este setat să ruleze la intervale regulate de timp. Acest mecanism asigură faptul că aplicația va actualiza periodic informațiile primite de la senzorii conectați. După inițierea cronometrajului, se verifică starea conexiunii Bluetooth prin apelul funcției BluetoothClient1.IsConnected. Această verificare este esențială, deoarece fără o conexiune activă, aplicația nu ar putea primi date. Dacă dispozitivul nu este conectat, restul procesului este oprit, evitându-se erorile ce ar putea apărea din încercarea de a primi date fără o conexiune validă.

Dacă conexiunea Bluetooth este activă, se trece la etapa următoare, care presupune verificarea disponibilității datelor pentru a fi primite. Aceasta se realizează prin funcția

BluetoothClient1.BytesAvailableToReceive, care returnează numărul de octeți (bytes) disponibili pentru a fi citați. În cazul în care sunt disponibili cel puțin 40 de octeți, aplicația va proceda la primirea datelor prin metoda BluetoothClient1.ReceiveText. Aceasta va salva datele primite într-o listă locală, care va conține valorile individuale provenite de la senzorii conectați. Fiecare element al listei va reprezenta o valoare diferită, cum ar fi pulsul, saturația de oxigen în sânge (SpO2), temperatura corpului, temperatura ambientală și umiditatea. Astfel, aplicația va putea manipula fiecare dată într-o manieră individuală și specifică.

Fig 5.14 Prima parte a programului



După ce datele au fost primite și stocate în listă, fiecare valoare este atribuită unor variabile separate. De exemplu, variabila puls va fi setată la valoarea aflată pe prima poziție din listă, corespunzătoare pulsului măsurat. Similar, spo2 va primi valoarea SpO2, iar tempc va reprezenta temperatura corpului. În același mod, alte variabile vor fi folosite pentru a stoca temperatura ambientală și umiditatea, obținute din alte poziții ale listei. Aceste variabile vor fi apoi folosite pentru a actualiza interfața utilizatorului, afișând valorile într-un format accesibil și ușor de citit pe dispozitivul mobil.

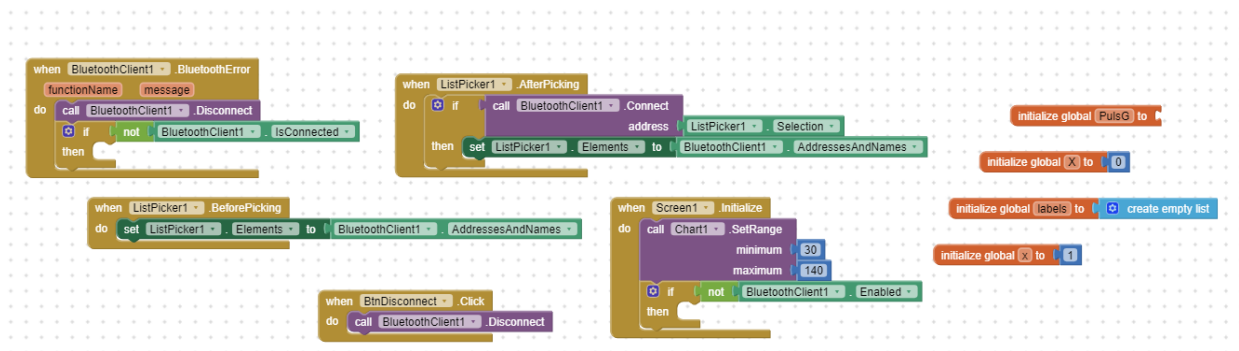
După ce valorile au fost extrase și stocate, aplicația va actualiza etichetele din interfața utilizatorului pentru a reflecta noile date. De exemplu, eticheta care reprezintă pulsul va fi setată să afișeze valoarea obținută de la senzorul de puls, iar etichetele pentru SpO2 și temperatură vor fi actualizate corespunzător. Astfel, utilizatorul poate observa în timp real valorile semnelor vitale ale pacientului sau ale utilizatorului dispozitivului. Pe lângă actualizarea etichetelor, aplicația va adăuga aceste date într-un grafic 2D, care urmărește evoluția parametrilor de-a

lungul timpului. Graficul va reflecta modificările pulsului și ale temperaturii pe măsură ce datele sunt primite și procesate, oferind o vizualizare clară a valorilor în timp.

De asemenea, aplicația include un mecanism de alertă care se activează atunci când temperatura depășește o valoare critică. Aceasta este o caracteristică importantă, care poate fi folosită pentru a alerta utilizatorul în cazul unor valori anormale, ce ar putea semnala o problemă medicală. În acest caz, aplicația va trimite o notificare simplă, cu un titlu și un subtitlu care indică faptul că temperatura pacientului este prea mare. Notificarea va fi afișată pe ecranul dispozitivului, asigurându-se că utilizatorul este informat imediat despre posibilele riscuri.

Pe măsură ce aplicația primește noi date de la senzorii conectați, trebuie să gestioneze și datele mai vechi. În acest scop, se implementează un mecanism de eliminare a valorilor mai vechi atunci când numărul acestora depășește o anumită limită (de exemplu, 10 valori). Astfel, aplicația va păstra doar cele mai recente date, asigurându-se că informațiile afișate sunt relevante și actuale. Aceste date vor fi utilizate pentru a actualiza continuu graficul și pentru a ajusta etichetele și notificările în funcție de noile informații disponibile.

Această primă parte a codului demonstrează modul în care aplicația primește, procesează și prezintă datele de la senzorii conectați, având ca scop monitorizarea semnelor vitale ale unui pacient sau ale unui utilizator. Procesul este bine structurat și asigură că datele sunt primite și prezentate în timp real, cu o interfață care oferă utilizatorului informații esențiale despre starea



acestuia.

Fig 5.15 A doua parte a programului

Capitolul 6. Testarea și rezultatele experimentale

În acest capitol, se va detalia procesul de testare al sistemului dezvoltat, precum și analiza rezultatelor obținute în urma experimentelor realizate. Testarea este o etapă esențială în orice proiect tehnologic, având rolul de a verifica funcționalitatea și fiabilitatea întregului sistem, de la hardware la software, și de a identifica eventualele îmbunătățiri necesare. În cadrul acestui capitol, vor fi prezentate etapele de testare efectuate, condițiile în care au fost realizate experimentele și evaluarea performanței dispozitivului.

De asemenea, vor fi prezentate datele colectate în timpul testelor, precum și analiza acestora, oferind o înțelegere detaliată asupra acurateței și preciziei sistemului în monitorizarea semnelor vitale. Testarea va include evaluarea principalelor funcționalități ale dispozitivului, cum ar fi măsurarea pulsului, a saturației de oxigen (SpO₂), a temperaturii corpului și a umidității, comparând rezultatele obținute cu valorile standard și cu dispozitive de referință. În plus, se va discuta despre provocările întâlnite în timpul testării și despre soluțiile implementate pentru optimizarea performanței sistemului.

Fig 6.1 Testarea senzorului de puls și a celui de temperatură corporală

Primul test efectuat a fost dedicat senzorului de puls, precum și senzorului de temperatură NTC, având ca scop verificarea acurateței măsurărilor de puls și temperatură corporală într-un cadru controlat. Procedura de testare a început prin conectarea senzorilor la placa de dezvoltare și configurarea inițială a sistemului, folosind codul dezvoltat pentru a prelua datele din acești senzori și a le vizualiza pe ecranul LCD al dispozitivului. Primul pas a fost verificarea conectivității între senzorii de puls și NTC cu restul sistemului, asigurându-ne că datele sunt transmise corect de la senzorii respectivi la microcontroler.

În cadrul testului, am monitorizat valorile obținute pentru puls și temperatura corpului în timp ce am simulat diverse condiții de măsurare, cum ar fi modificări ale temperaturii ambientale și ale ritmului cardiac. De exemplu, am măsurat pulsul la diferite intervale de timp, la repaus și după un efort fizic moderat, pentru a evalua reacția senzorilor la schimbările ritmului cardiac. De asemenea, temperatura corporală a fost monitorizată în timp ce subiectul testului rămânea într-un mediu cu temperatură controlată, pentru a observa eventualele fluctuații ale măsurărilor.

Unul dintre primele probleme întâmpinate a fost o diferență semnificativă între valorile obținute de la senzorul NTC și valorile măsurate cu un termometru de referință. În mod specific, senzorul NTC părea să returneze valori mai scăzute decât cele reale, ceea ce ar fi putut duce la o eroare în calculul temperaturii corpului. După o analiză detaliată a codului și a conexiunilor senzorilor, am descoperit că eroarea era cauzată de o calibrare incorectă a senzorului NTC, care nu ținea cont de rezistența inițială a senzorului la temperatura de referință.

Pentru a rezolva această problemă, am efectuat o recalibrare a senzorului NTC, aplicând formula corectă pentru a compensa variațiile inițiale ale rezistenței senzorului în raport cu temperatura.

De asemenea, am implementat un algoritm suplimentar în codul programului pentru a ajusta citirile senzorului în funcție de temperatura de referință măsurată cu termometrul extern. Astfel, am reușit să îmbunătățim acuratețea măsurătorilor de temperatură corporală, iar după această ajustare, valorile obținute de la senzorul NTC au fost mult mai apropiate de valorile reale.

În ceea ce privește senzorul de puls, am constatat că acesta funcționa corespunzător în majoritatea condițiilor de testare, însă la anumite intervale de măsurare, senzorul tindea să dea citiri instabile, mai ales atunci când subiectul se mișca brusc sau când conexiunile nu erau suficient de stabile. Acest comportament putea fi explicat prin interferențele electromagnetice sau prin contactul insuficient al senzorului cu pielea. Pentru a rezolva această problemă, am îmbunătățit fixarea senzorului pe piele și am realizat teste suplimentare pentru a verifica stabilitatea semnalului pe o perioadă mai lungă de timp. De asemenea, am adăugat un algoritm de filtrare a semnalului în cod, care a ajutat la eliminarea fluctuațiilor bruște și a îmbunătățirii stabilității măsurătorilor.

În concluzie, testarea senzorilor de puls și temperatură a fost un pas esențial în evaluarea sistemului, iar problemele întâmpinate au fost rezolvate prin ajustarea calibrării senzorilor și îmbunătățirea stabilității măsurătorilor. Rezultatele obținute după aceste ajustări au fost mult mai precise și au permis continuitatea testării și evaluării altor componente ale sistemului. Aceste experiențe ne-au oferit o înțelegere mai profundă a limitărilor tehnologiei utilizate și au cont

Fig 6.2 Testarea ecranului și a senzorului DHT

Al doilea test realizat a fost dedicat senzorului DHT11 și modulului LCD, având ca scop verificarea corectitudinii și stabilității măsurătorilor de temperatură și umiditate, dar și a afișării acestora pe ecranul LCD. Procedura de testare a început prin conectarea senzorului DHT11 la microcontroler și configurarea programului astfel încât datele să fie preluate corect de la senzor și afișate pe ecranul LCD. Inițial, am verificat conectivitatea dintre senzorul DHT11 și microcontroler, asigurându-ne că semnalele de temperatură și umiditate erau transmise corect. În acest sens, am monitorizat valorile citite pe terminalul serial pentru a ne asigura că senzorul transmitea date valide.

Testarea a continuat prin simularea diferitelor condiții de mediu, cum ar fi modificarea temperaturii ambientale și a umidității, pentru a observa comportamentul senzorului în fața acestor variabile. De exemplu, am testat senzorul într-un mediu controlat, la temperaturi scăzute și ridicate, iar ulterior am măsurat umiditatea prin adăugarea de vapori de apă în aerul înconjurător. În același timp, am verificat modul în care datele erau afișate pe ecranul LCD, pentru a ne asigura că valorile de temperatură și umiditate erau prezentate în mod corespunzător și că display-ul reacționa la modificările acestor parametri.

În timpul acestor teste, însă, am întâmpinat o problemă legată de sincronizarea senzorului DHT11 cu microcontrolerul. Deși senzorul trimitea datele, am observat că valorile de temperatură și umiditate nu erau actualizate corect sau erau afișate pe ecranul LCD la intervale

neregulate. După o analiză atentă a codului, am descoperit că eroarea era cauzată de o problemă de sincronizare între senzor și microcontroler. Senzorul DHT11 necesită un interval precis de timp între citirile succesive pentru a furniza date corecte, iar în cazul în care acest interval nu era respectat, valorile afișate pe LCD erau incomplete sau incorecte.

Pentru a rezolva această problemă, am ajustat programul pentru a respecta intervalele corecte de citire a datelor de la senzorul DHT11. Am introdus o întârziere de câteva secunde între citirile succesive ale senzorului, astfel încât să fie asigurată o sincronizare adecvată. De asemenea, am implementat o metodă de verificare a valabilității datelor citite de la senzor, astfel încât să fie evitate citirile eronate, ce ar putea duce la afișarea unor valori incorecte pe LCD. După aceste ajustări, senzorul a început să furnizeze date corecte și actualizate constant, iar ecranul LCD a început să afișeze valorile de temperatură și umiditate într-un mod stabil și continuu.

Un alt aspect important pe care l-am testat a fost vizibilitatea informațiilor pe ecranul LCD, având în vedere că modulul utilizat era un display I2C de 20x4 caractere. În unele condiții de iluminare mai slabă, am observat că textul nu era vizibil suficient de clar. Pentru a îmbunătăți această situație, am ajustat luminozitatea display-ului și am optimizat poziționarea textului pe ecran astfel încât valorile de temperatură și umiditate să fie mai ușor de citit. De asemenea, am implementat un sistem de contrast ajustabil, care permite utilizatorului să modifice vizibilitatea ecranului în funcție de condițiile de iluminare ambientale.

În concluzie, testarea senzorului DHT11 și a modulului LCD a fost esențială pentru validarea funcționalității sistemului în ceea ce privește măsurarea și afișarea datelor de temperatură și umiditate. Problemele întâmpinate au fost soluționate prin ajustarea codului pentru a respecta intervalele corecte de citire ale senzorului și prin optimizarea vizibilității datelor pe ecranul LCD. După aceste îmbunătățiri, senzorul a funcționat conform așteptărilor, iar sistemul a demonstrat o stabilitate mai mare în măsurători și afișare, oferind un instrument fiabil pentru monitorizarea condițiilor de mediu.

Fig 6.3 Testarea aplicației mobile

Al treilea test realizat a fost dedicat aplicației Bluetooth și notificărilor pentru temperatura corpului, cu scopul de a verifica funcționalitatea sistemului de monitorizare a semnelor vitale, în special a temperaturii corpului, prin intermediul notificărilor trimise către utilizator în cazul în care această valoare depășește anumite limite prestabilite. Procedura de testare a început cu configurarea aplicației Bluetooth pe dispozitivul mobil, iar în paralel am efectuat conexiunea între aplicația mobilă și microcontrolerul care gestionează senzorul de temperatură. Am verificat apoi conectivitatea Bluetooth între cele două dispozitive, asigurându-mă că datele de la senzorul de temperatură erau transmise corect prin Bluetooth către aplicația mobilă, iar aplicația putea să primească și să afișeze aceste date într-un mod corespunzător.

Testul a continuat prin simularea mai multor condiții în care temperatura corpului ar putea varia, astfel încât să pot verifica reacția aplicației mobilă la modificările de temperatură. În acest sens, am monitorizat comportamentul aplicației în timp ce am măsurat temperatura corpului folosind senzorul corespunzător, iar ulterior am observat cum aplicația trimite notificări în cazul în care

temperatura depășea pragurile stabilite de siguranță, indicând o posibilă stare febrilă. De asemenea, am testat și funcționalitatea notificărilor, astfel încât utilizatorul să poată primi alerte vizibile pe ecranul dispozitivului atunci când temperatura corpului crește peste valoarea normală.

În timpul acestor teste, însă, am întâmpinat o problemă legată de sincronicitatea între aplicația Bluetooth și notificările generate de aplicația mobilă. În anumite condiții, notificările nu erau trimise la timp, chiar dacă temperatura depășea pragul stabilit. Aceasta se întâmpla atunci când aplicația mobilă nu reușea să preia și să proceseze datele de temperatură în timp real, iar notificările erau întârziate sau nu se activau deloc. După investigarea detaliată a cauzei, am descoperit că problema era legată de gestionarea corespunzătoare a conexiunii Bluetooth, deoarece aplicația nu actualiza în mod continuu și rapid datele primite de la senzor.

Pentru a rezolva această problemă, am refăcut fluxul de date al aplicației pentru a asigura o sincronizare mai rapidă și mai eficientă între senzorul de temperatură, microcontroler și aplicația mobilă. Am optimizat algoritmul care preia datele de la senzor astfel încât aplicația să fie capabilă să actualizeze permanent valorile de temperatură și să proceseze notificările într-un interval de timp scurt. De asemenea, am implementat o verificare mai riguroasă a stării conexiunii Bluetooth pentru a preveni situațiile în care datele nu ajungeau corect la aplicație din cauza pierderii temporare a conexiunii.

În plus, am testat și îmbunătățit modul în care sunt gestionate notificările, astfel încât să fie trimise imediat ce temperatura corpului depășește pragul setat. Am inclus o logică de verificare continuă a valorii de temperatură, iar atunci când aceasta se află în afacerea limitei normale, aplicația trimite o notificare vizibilă utilizatorului. Am inclus și un sistem de notificare sonoră pentru a atrage atenția utilizatorului asupra riscului potențial. După ajustările efectuate, am verificat funcționarea aplicației într-un mediu real, în care am testat atât conectivitatea Bluetooth cât și activarea notificărilor la diferite valori de temperatură. Rezultatele obținute au fost foarte bune, iar aplicația a început să funcționeze într-un mod optim, trimite notificări în timp util și fără întârzieri.

În concluzie, testarea aplicației Bluetooth și a notificărilor pentru temperatura corpului a fost esențială pentru a asigura o funcționare corectă a sistemului de monitorizare a stării de sănătate. Problemele de sincronizare dintre aplicație și senzor au fost rezolvate prin optimizarea fluxului de date și a algoritmului de prelucrare a acestora, iar aplicația a început să trimită notificări instantanee atunci când temperatura corpului depășea pragurile normale. După aceste ajustări, aplicația s-a dovedit a fi fiabilă și eficientă, asigurând un sistem de monitorizare a sănătății care poate fi utilizat într-un mod sigur și util pentru utilizatori.

În urma testării aplicate pentru sistemul de monitorizare a semnelor vitale, am identificat atât puncte forte, cât și puncte slabe care au influențat performanța generală a sistemului. În primul rând, un aspect pozitiv important a fost stabilitatea conexiunii Bluetooth între dispozitivul mobil și senzorii utilizați, ceea ce a asigurat o transmitere fiabilă a datelor, fără pierderi semnificative de informații. De asemenea, aplicația a reușit să proceseze și să afișeze valorile măsurate (precum pulsul, SpO2 și temperatura) într-un mod eficient și fără întârzieri notabile, iar notificările trimise în cazul în care temperatura corpului depășea limitele stabilite au funcționat corespunzător, contribuind la realizarea unui sistem de alertare adecvat. De asemenea, interfața utilizatorului a fost clară și ușor de utilizat, iar graficele afișate pentru evoluția valorilor măsurate

au oferit o vizualizare intuitivă a datelor. Aceste rezultate pozitive sugerează că sistemul este capabil să răspundă rapid la schimbările din parametrii vitali ai pacientului și poate oferi informații utile în timp real.

Cu toate acestea, au existat și unele puncte slabe care au fost evidente pe parcursul testării. Unul dintre principalele probleme identificate a fost legată de sincronizarea între măsurătorile realizate de senzorii de temperatură și aplicația mobilă. Deși senzorii de temperatură au funcționat corect în majoritatea cazurilor, uneori aplicația a întâmpinat dificultăți în procesarea datelor în timp real. Aceasta a dus la o întârziere în actualizarea valorilor și, în anumite cazuri, la notificări care nu erau trimise imediat ce temperatura corpului depășea pragul normal. De asemenea, în anumite condiții, senzorul de temperatură NTC nu oferea o valoare suficient de precisă, ceea ce a afectat corectitudinea măsurătorilor. Aceste întârzieri și erori de măsurare au fost cauzate în principal de limitările legate de performanța hardware-ului folosit, dar și de unele aspecte legate de gestionarea eficientă a fluxului de date în aplicația mobilă.

Pentru a îmbunătăți performanța sistemului pe viitor, este esențial să se abordeze aceste probleme de sincronizare și precizie. În primul rând, ar fi necesară optimizarea algoritmilor care prelucrează datele din senzorii de temperatură și implementarea unui sistem mai eficient de gestionare a fluxului de date în timp real. De asemenea, s-ar putea lua în considerare utilizarea unui senzor de temperatură mai precis, cu o gamă de măsurare mai largă și o stabilitate mai bună în condiții de temperaturi extreme sau fluctuații rapide de temperatură. Îmbunătățirea performanței senzorilor ar putea contribui semnificativ la creșterea acurateții datelor și la reducerea erorilor de măsurare.

Un alt punct de îmbunătățire ar putea fi legat de durata de viață a bateriei dispozitivelor care utilizează senzorii de măsurare, mai ales în contextul utilizării acestora pe termen lung. În testele efectuate, am observat că bateria se descărca rapid atunci când sistemul era folosit continuu pentru monitorizarea semnelor vitale, iar acest aspect ar putea limita aplicabilitatea dispozitivului într-un cadru clinic sau în situații de urgență. Pentru a rezolva această problemă, ar putea fi implementate strategii de economisire a energiei, cum ar fi oprirea senzorilor atunci când nu sunt necesari sau implementarea unor module de alimentare mai eficiente din punct de vedere energetic.

Un alt aspect care poate fi îmbunătățit este gestionarea notificărilor și a alertei sonore. Deși notificările au fost trimise corect, uneori utilizatorul nu a avut suficient timp să reacționeze la notificările trimise din cauza lipsei unui feedback mai imediat sau mai semnificativ. În viitor, ar fi util să se implementeze o alertă mai proeminentă, poate chiar cu vibrații sau sunete specifice, care să atragă atenția utilizatorului mai rapid. De asemenea, ar fi posibilă includerea unei funcționalități de confirmare a primirii notificării, pentru a asigura că utilizatorul este conștient de schimbările semnificative ale parametrilor de sănătate.

Pe lângă aceste aspecte, un alt punct ce ar putea fi îmbunătățit ar fi legat de afișarea graficelor pe ecranul aplicației. În anumite condiții, graficele nu reflectau cu exactitate evoluția valorilor în timp, din cauza unor erori de sincronizare între datele colectate de la senzori și actualizarea graficului. O soluție posibilă ar fi optimizarea algoritmului care actualizează graficele, asigurându-se că datele sunt procesate și afișate în mod continuu și corect, fără întârzieri sau erori de interpretare.

În concluzie, deși testele realizate au evidențiat atât aspecte pozitive, cât și provocări semnificative, aceste probleme pot fi rezolvate prin actualizări software și hardware, optimizări ale performanței senzorilor și îmbunătățirea modului de gestionare a fluxului de date. Continuarea cercetărilor și implementarea unor soluții tehnice inovatoare vor contribui la crearea unui sistem mai robust și mai fiabil, care poate fi utilizat eficient în monitorizarea stării de sănătate a pacienților, atât acasă, cât și în spitale.

Concluzii

În cadrul acestei lucrări, am urmărit dezvoltarea unui sistem complex de monitorizare a semnelor vitale, care să integreze mai mulți senzori pentru a colecta date referitoare la puls, saturația de oxigen în sânge (SpO2), temperatura corpului și umiditatea mediului. Scopul principal al acestei lucrări a fost crearea unui dispozitiv medical care să permită măsurarea acestor parametri vitali într-un mod eficient și să ofere informații relevante pentru evaluarea stării de sănătate a utilizatorului. Proiectul a vizat nu doar colectarea datelor de la senzori, dar și integrarea acestora într-o aplicație mobilă, prin intermediul unei interfețe de utilizator clare și intuitive, care să permită vizualizarea și analiza în timp real a datelor măsurate. De asemenea, am urmărit implementarea unui sistem de notificări care să alerteze utilizatorul în caz de valori anormale, contribuind astfel la prevenirea unor posibile probleme de sănătate.

Lucrarea a avut un parcurs semnificativ în ceea ce privește realizarea obiectivelor propuse. Am reușit să construiesc un sistem care utilizează senzorii de puls, SpO2, temperatură (NTC și DHT11), precum și un modul Bluetooth pentru a transmite datele către aplicația mobilă, care le prelucrează și le afișează într-un format ușor de înțeles. În cadrul testelor realizate, am observat o bună sincronizare a senzorilor și o transmitere eficientă a datelor prin Bluetooth, iar aplicația a reușit să afișeze valorile măsurate într-un mod coerent. De asemenea, notificările pentru valori anormale de temperatură au funcționat corect, iar graficele pentru monitorizarea în timp real a evoluției semnelor vitale au oferit o vizualizare clară și utilă a datelor colectate. Totodată, am întâmpinat unele dificultăți legate de precizia senzorilor și de sincronizarea datelor, care au fost identificate și discutate, iar soluțiile propuse au vizat îmbunătățirea performanței sistemului prin optimizarea codului și selectarea unor componente mai precise.

Un alt aspect important al lucrării a fost integrarea tuturor componentelor hardware și software într-un sistem unitar. Am lucrat cu mai multe platforme și tehnologii, inclusiv Arduino, pentru a conecta senzorii la microcontroler și a le permite să comunice cu aplicația mobilă prin Bluetooth. Am implementat diverse funcționalități software pentru a gestiona fluxul de date și pentru a crea o interfață de utilizator care să ofere o experiență intuitivă și eficientă. De asemenea, am colaborat cu diverse biblioteci și module externe pentru a facilita comunicarea cu senzorii și pentru a obține măsurători precise.

Contribuțiile personale în acest proiect au fost esențiale pentru realizarea acestuia. Am fost responsabil pentru dezvoltarea și testarea codului, selectarea senzorilor, realizarea conexiunilor hardware și implementarea funcționalităților software pentru aplicația mobilă. De asemenea, am realizat testele necesare pentru a evalua performanța sistemului și pentru a identifica posibilele îmbunătățiri. În timpul dezvoltării, am învățat mult despre gestionarea fluxului de date în timp

real și despre provocările întâlnite în integrarea senzorilor și a aplicațiilor mobile. Am învățat, de asemenea, importanța testării riguroase și a identificării rapide a problemelor, ceea ce mi-a permis să optimizez sistemul și să propun soluții de îmbunătățire.

În ceea ce privește perspectivele viitoare, acest proiect poate fi extins prin integrarea unor senzori suplimentari, care să permită monitorizarea unor parametri suplimentari de sănătate, cum ar fi tensiunea arterială sau electrocardiograma. De asemenea, se pot îmbunătăți performanțele senzorilor, prin utilizarea unor componente de înaltă precizie și mai fiabile, care să contribuie la creșterea acurateții măsurărilor. În plus, aplicația mobilă ar putea fi îmbunătățită prin adăugarea unor funcționalități suplimentare, cum ar fi stocarea și analiza istoricului datelor, crearea de rapoarte periodice și integrarea cu platforme de telemedicină pentru monitorizarea de la distanță a pacienților.

Bibliografie

1. Aarts, René, and Harry L. Bunt. *Introduction to Medical Technologies*. Springer, 2016.
2. Al-Fuqaha, Ala, et al. "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications." *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, 2015, pp. 2347–2376.
3. Basso, Cosimo, and Stefano Ferretti. "Wireless Sensor Networks for Monitoring Vital Signs in Healthcare Applications." *Springer Handbook of Automation*, Springer, 2016, pp. 1095-1126.
4. Brusey, James, and Michael M. M. B. "Wearable Healthcare Devices: Technologies and Applications." *Springer Handbook of Electronics for Engineers*, Springer, 2017.
5. Chien, Yung-Kai, et al. "A Mobile Cloud-Based Real-Time Health Monitoring System with Internet of Things." *Sensors*, vol. 17, no. 12, 2017, p. 2764.
6. Dahiya, Ravinder, et al. *Smart Sensors for Health and Environment Monitoring*. CRC Press, 2020.
7. Ekberg, Eric, and Ivan Fleiszer. "Telemedicine and Its Application in Healthcare." *Telemedicine and e-Health*, vol. 25, no. 6, 2019, pp. 482-491.
8. Hossain, M. Shahin, et al. "Wireless Body Area Networks (WBANs) for Healthcare Applications." *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, 2014, pp. 2180–2187.
9. Huang, Qian, et al. "Smart Health: A Review of Technologies and Applications." *IEEE Access*, vol. 8, 2020, pp. 90442-90455.
10. Jovanov, Emil, et al. "Wireless Health Monitoring: The Sensor Network Perspective." *IEEE Transactions on Information Technology in Biomedicine*, vol. 8, no. 1, 2004, pp. 1-11.
11. Kamel Boulos, M. N., et al. "Wearable Sensors in Medical Internet of Things: Current Advances and Future Directions." *Journal of Healthcare Engineering*, vol. 2020, 2020, p. 7389475.
12. Kim, Yong-Sang, et al. "Smart Healthcare Monitoring Using Wearable Wireless Medical Devices." *Health Information Science and Systems*, vol. 3, no. 1, 2015, pp. 1-8.
13. Li, Chia-Hsiu, et al. "Real-time Patient Monitoring System in Healthcare using Internet of Things (IoT)." *Computers in Biology and Medicine*, vol. 94, 2018, pp. 121-129.

14. Liu, Bing, and Guo-Wei Yu. *IoT for Healthcare Applications and Devices*. CRC Press, 2021.
15. Lu, Weiliang, et al. "A Wearable IoT-Based System for Continuous Monitoring of Cardiovascular Diseases." *IEEE Internet of Things Journal*, vol. 7, no. 10, 2020, pp. 9514–9523.
16. Nguyen, Huong Thanh, et al. "A Survey on Internet of Things (IoT) for Healthcare Systems." *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 6, 2019, pp. 206-213.
17. Patel, S., et al. "Wearable Health Monitoring Systems: A Review." *IEEE Sensors Journal*, vol. 13, no. 6, 2013, pp. 2169-2175.
18. Ren, Li, et al. "Internet of Things and Healthcare Applications." *Springer Handbook of Internet of Things*, Springer, 2016, pp. 439-456.
19. Stojanovic, Jovan, et al. "Wearable Health Monitoring Systems for Continuous Care." *Biomedical Engineering Applications, Basis & Communications*, vol. 31, no. 7, 2019, pp. 85-99.
20. Tiwari, Puneet, et al. "A Review of Smart Sensors for Healthcare Applications." *Sensors*, vol. 19, no. 9, 2019, p. 2171.
21. Wu, Lijun, et al. "A Review on Smart Sensors and Applications in Healthcare Systems." *Sensors*, vol. 19, no. 19, 2019, p. 4163.
22. Zhang, Ying, and Wei Xu. "An IoT-Based Healthcare System for Patient Monitoring." *Journal of Biomedical Informatics*, vol. 65, 2017, pp. 95–104.
23. Zheng, Kai, et al. "Smart Healthcare Systems and Wearable Devices: A Review." *Sensors*, vol. 21, no. 7, 2021, p. 2509.

Anexe

Anexa 1 Codul sursă al sistemului

```
#include "DHT.h"

#define DHTPIN D5 // Pin digital conectat la senzorul DHT

#define DHTTYPE DHT11 // DHT 11


#include <Wire.h>

#include <LiquidCrystal_I2C.h>

#include <SoftwareSerial.h>

SoftwareSerial bt(D7, D6); // RX, TX

// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 20, 4);

DHT dht(DHTPIN, DHTTYPE);


#include <Wire.h>

#include "MAX30100_PulseOximeter.h"


#define PERIOADA_RAPORTARE_MS 1000

PulseOximeter pox;

#define RT0 10000 // Ω

#define B 3977 // K

float a=A0;
```

```

#define VCC 5 //Supply voltage

#define R 10000 //R=10KΩ

//Variables

float RT, VR, ln, Temp, T0, Read;

uint32_t tsUltimRaport = 0;

void setup()

{

    Serial.begin(115200);

    lcd.begin();

    // Turn on the backlight and print a message.

    lcd.backlight();

    dht.begin();

    bt.begin(9600);

    if (!pox.begin()) {
Serial.println("FAILED");

for(;;);

    } else {

Serial.println("SUCCESS");

    }

    pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);

```

```

    pinMode(A0,INPUT);

    T0 = 25 + 273.15;

}

void loop()

{

    pox.update();

    if (millis() - tsUltimRaport > PERIOADA_RAPORTARE_MS) {

        int ritmCardiac = pox.getHeartRate();

        float umiditate = dht.readHumidity();

        float temperatura = dht.readTemperature();

        Serial.print("Ritm cardiac: ");

        Serial.print(ritmCardiac);

        Serial.print(" bpm / SpO2: ");

        lcd.setCursor(0,0);

        lcd.print("Puls:");

        lcd.print(ritmCardiac);

        lcd.print("BPM");

        lcd.setCursor(0,1);

        lcd.print("Spo2:");

```

```

lcd.print(pox.getSpO2());

lcd.setCursor(7,1);

lcd.print("%");


Read = analogRead(A0);          //Acquisition analog value Read

Read = (5.00 / 1023.00) * Read;  //Conversion to voltage

VR = VCC - Read;

RT = Read / (VR / R);           //Resistance of RT


ln = log(RT / RT0);

Temp = (1 / ((ln / B) + (1 / T0))); //Temperature from sensor


Temp = Temp - 273.15;           //Conversion to Celsius

lcd.setCursor(0,2);

lcd.print("T:");

lcd.print(temperatura);

lcd.print("C ");

lcd.print("H:");

lcd.print(umiditate);

lcd.print("%");

lcd.setCursor(0,3);

lcd.print("TempCorp:");

lcd.print(Temp);

```



```
    lcd.print("C");  
  
    bt.print(ritmCardiac);  
  
    bt.print(";");  
  
    bt.print(pox.getSpO2());  
  
    bt.print(";");  
  
    bt.print(Temp);  
  
    bt.print(";");  
  
    bt.print(temperatura);  
  
    bt.print(";");  
  
    bt.print(umiditate);  
  
    bt.println(";");  
  
    Serial.print(pox.getSpO2());  
  
    Serial.println("% ");  
  
    tsUltimRaport = millis();  
  
}  
  
}
```

**DECLARAȚIE DE AUTENTICITATE A
LUCRĂRII DE FINALIZARE A STUDIILOR***

Subsemnatul DINIȚĂ ROXANA DANIELA

legitimată cu C.I. seria T2 nr. 679720

CNP 299 0430 250028

autorul lucrării Monitorizarea la Distanță a Parametrilor Vitali prin IoT și Aplicații
Mobile

elaborată în vederea susținerii examenului de finalizare a studiilor de
Licență organizat de către Facultatea

Automatică și Calculatoare din cadrul Universității
Politehnica Timișoara, sesiunea ianuarie - februarie a anului universitar

2024-2025, coordonator MIHAELA CRISAN VIDA, luând în

considerare art. 34 din Regulamentul privind organizarea și desfășurarea examenelor de
licență/diplomă și disertație, aprobat prin HS nr. 109/14.05.2020 și cunoscând faptul că în
cazul constatării ulterioare a unor declarații false, voi suporta sancțiunea administrativă
prevăzută de art. 146 din Legea nr. 1/2011 – legea educației naționale și anume anularea
diplomei de studii, declar pe proprie răspundere, că:

- această lucrare este rezultatul propriei activități intelectuale;
- lucrarea nu conține texte, date sau elemente de grafică din alte lucrări sau din alte
surse fără ca acestea să nu fie citate, inclusiv situația în care sursa o reprezintă o altă
lucrare/alte lucrări ale subsemnatului;
- sursele bibliografice au fost folosite cu respectarea legislației române și a convențiilor
internationale privind drepturile de autor;
- această lucrare nu a mai fost prezentată în fața unei alte comisii de examen/prezentată
public/publicată de licență/diplomă/disertație;
- În elaborarea lucrării am utilizat instrumente specifice inteligenței artificiale (IA) și anume
ChatGPT (denumirea OpenAI) (sursa), pe care le-am citat în conținutul lucrării/nu am utilizat
instrumente specifice inteligenței artificiale (IA)¹.

Declar că sunt de acord ca lucrarea să fie verificată prin orice modalitate legală pentru
confirmarea originalității, consimțind inclusiv la introducerea conținutului său într-o bază de
date în acest scop.

Timișoara,

Data

26.06.2025

Semnătura



*Declarația se completează de student, se semnează olograf de acesta și se inserează în lucrarea de
finalizare a studiilor, la sfârșitul lucrării, ca parte integrantă.

¹ Se va păstra una dintre variante: 1 - s-a utilizat IA și se menționează sursa 2 – nu s-a utilizat IA

