

University of Business and Technology in Kosovo

UBT Knowledge Center

Theses and Dissertations

Student Work

Winter 12-2020

Analizimi dhe zhvillimi i sistemit për komunikim në telemjekësi duke përdorur teknologjinë webRTC

Fitim Mehmeti

Follow this and additional works at: <https://knowledgecenter.ubt-uni.net/etd>



Part of the [Computer Sciences Commons](#)



Programi për Shkenca Kompjuterike dhe Inxhinierise

**Analizimi dhe zhvillimi i sistemit për komunikim në telemjekësi duke
përdorur teknologjinë webRTC**
Shkalla Bachelor

Fitim Mehmeti

Dhjetor /2020
Prishtinë



Programi për Shkenca Kompjuterike dhe Inxhinierise

Punim Diplome
Viti akademik 2011 – 2012

Fitim Mehmeti

**Analizimi dhe zhvillimi i sistemit për komunikim në telemjekësi duke
përdorur teknologjinë webRTC**

Mentori: Can.PhD. Xhelal Jashari

Dhjetor /2020

Ky punim është përpiluar dhe dorëzuar në përmbushjen e kërkesave të
pjeshme për Shkallën Bachelor

ABSTRAKT

Sistemet softuerike për komunikim në përgjithësi e në veçanti telemedicina që për qëllim ka ofrimin e kujdesit shëndetsor në distancë janë duke u bërë aset i rëndësishëm në shtetet e zhvilluara. Kufizimet në kohë, vendndodhja, kushtet atmosferike, tash së fundmi edhe pandemia COVID-19 janë ndër arsyet kryesore për zhvillimin e sistemeve të tilla.

Ky punim është përqëndruar në hulumtimin, analizimin dhe krahasimin e sistemeve për komunikim online, mënyra se si funksionojnë këto sisteme, gjithashtu është fokusuar në zhvillimin e një aplikacioni për komunikim me video për pacientët dhe mjekët duke përdorur teknologjinë webRTC, ky projekt do të zhvillohet në bazë të kërkesave të një spitali në tregun tonë.

Në këtë temë po ashtu kemi përshkruar dhe implementuar veçoritë standarde që i posedojnë këto lloj të sistemeve, si dhe kemi analizuar funksione të reja për t'i integruar në të ardhmen në këtë sistem.

***Fjalët Kyçe:** Sistemet për komunikim, Telemedicina, webRTC.*

MIRËNJOHJE/FALENDERIME

I falemnderohem Zotit për shëndetin dhe mirëqenien që më percollën gjatë përgaditjes së këtij punimi si dhe gjatë gjithë kohëzgjatjes së studimeve. Mbështetja e vazhdueshme dhe e pakursyer morale dhe financiare nga prindërit ka qenë gjithmonë motivimi kryesor për realizimin e synimeve të mia.

Falënderoj përzemërsisht prindërit, vëllaun dhe motërat si dhe të gjithë familjen për përkrahjen, këshillat dhe respektin që më kanë ofruar.

Gjithashtu falënderime të sinqerta shpreh për shoqërinë dhe kolegët e mi të cilët kanë qenë gjithmonë të gatshëm për bashkëpunim dhe për diskutime rreth çdo problemi që kam hasur gjatë studimeve të mia.

Falënderim dhe mirënjohje për gjithë stafin e fakultetit të kolegjit UBT, në veçanti profesorëve të departementit të shkencave kompjuterike të cilit unë i përkas. Në mënyrë të veçantë falënderoj Prof. Xhelal Jashari, për kohën, gatishmërinë dhe përkrahjen e rezervë gjatë punimit të temës së diplomës.

Të gjithë atyre që në mënyrë të drejtpërdrejtë ose tërthorazi kanë kontribuar në përfundimin e kësaj sfide, ju falemnderit!

PËRMBAJTJA

1	HYRJE	1
2	SHQYRTIMI I LITERATURËS	3
2.1	Sistemet për komunikim	3
2.2	Telemjekësia dhe komponentët e saj	4
2.3	Analizimi i Sistemeve për komunikim	6
2.3.1	Doxy.me	6
2.3.2	eVisit.....	7
2.3.3	Teladoc	9
2.3.4	Krahasimi i sistemeve.....	10
3	DEKLARIMI I PROBLEMIT	12
4	METODOLGJIA	13
5	REZULTATET	14
5.1	Përshkrimi i moduleve të sistemit	15
5.1.1	Kyçja në sistem.....	15
5.1.2	Moduli i adminit për stafin mjeksorë	17
5.1.3	Moduli i pacientit.....	29
5.1.4	Moduli i doktorit.....	31
5.2	Teknologjitë e përdorura	36
5.2.1	Teknologjia webRTC	37
5.2.2	WebSocket.....	40
5.3	Gjuhët Programuese	42
5.3.1	HTML.....	42
5.3.2	CSS	42
5.3.3	PHP	42
5.3.4	Javascript	43
5.3.5	Baza e të dhënave	44
5.4	Kërkesat e sistemit.....	45
5.4.1	Kërkesat funksionale	45
5.4.2	Kërkesat jo-funksionale	45
6	DISKUTIME DHE PËRFUNDIME	46
7	REFERENCAT	47

Lista e Figurave

Figura 1. Topologjia e video komunikimit.....	4
Figura 2. Sistemi Doxy.me.....	6
Figura 3. Doxy.me Login.....	7
Figura 4. Sistemi eVisit.....	8
Figura 5. eVisit Login.....	8
Figura 6. Sistemi Teladoc.....	9
Figura 7. Teladoc Login.....	10
Figura 8. Login për përdoruesin.....	15
Figura 9. Dashboard për stafin mjeksorë.....	18
Figura 10. Lista e pacientëve për stafin mjeksorë.....	19
Figura 11. Faqja për shtimin e një pacienti.....	20
Figura 12. Faqja për të dhënat e pacientit.....	22
Figura 13. Lista e doktorëve.....	23
Figura 14. Faqja për shtimin e një doktori.....	24
Figura 15. Faqja për të dhënat e doktorit.....	26
Figura 16. Lista e departamenteve.....	27
Figura 17. Faqja për shtimin e departamentit.....	28
Figura 18. Dashboard për pacientin.....	30
Figura 19. Terminet për pacientin.....	31
Figura 20. Dashboard për doktorin.....	32
Figura 21. Terminet për doktorin.....	33
Figura 23. Trekendeshi webRTC.....	37
Figura 24. Shkëmbimi i mesazheve mes klientëve duke përdorur serverin për sinjalizim ..	38
Figura 25. Klientet e pranojnë ip adresen publike nepermjet serverit STUN.....	39
Figura 26. Te dhenat e dy klienteve barten nepermjet serverit TURN.....	40
Figura 27. Komunikimi në websocket.....	41

LISTA E TABELAVE

Tabela 1. Përshkrimi i sistemeve.....	11
---------------------------------------	----

FJALORI I TERMAVE

HTML	Hypertext Markup Language
CSS	Cascading Style Sheet
WEBRTC	Web Real-Time communication
ICE	Interactive Connectivity Establishment
TURN	Traversal Using Relays around NAT
STUN	Session Traversal for NAT
NAT	Network address translation
SDP	Session Description Protocol
FTP	File Transfer Protocol
TCP	Transmission Control Protocol
HTTP	Hypertext Transfer Protocol
PHP	Hypertext Preprocessor
IP	Internet Protocol
API	Application Programming Interface
SQL	Structured Query Language
SaaS	Software as a Service

1 HYRJE

Interneti është kudo në shoqërinë e sotme, ai mund të gjendet në shtëpi, kafene, autobusë e madje viteve të fundit edhe në celularët tanë. Ai ka krijuar një mënyrë krejt të re të komunikimit mes njerëzve në të gjithë botën. Nuk ka rëndësi se ku jeni qasja në internet është pothuajse në dispozicion në të gjithë vendet.

Bashkë me qasje në internet vjen edhe mundësia për të komunikuar me njerëz të tjerë. Në ditët e sotme celularët janë bërë paisjet më të përdorurura për qasje në internet dhe pjesë e domosdoshme e jetës sonë.

Edhe pse interneti fillimisht shërbeu për publikimin e përmbajtjes statike, aplikacionet web dhe ato mobile përdoren me të madhe për komunikim në internet. Këto aplikacione me teknologjinë në “kohë-reale” bëjnë të mundshme komunikimin nga serveri tek klientët, pra në vend që vetëm klientët të dërgojnë kërkesa në server, me anë të këtyre aplikacioneve serveri mund të pajisë të gjithë klientët e lidhur me informacione posa ato të jenë në dispozicion.

Teknologjitë në kohë reale mundësuan krijimin e aplikacioneve për komunikim multimedial direkt në paisjet kompjuterike, ku freskia e të dhënave është shumë e rëndësishme.

Në të kaluarën për krijimin e aplikacioneve në kohë reale janë përdorur shtojca të cilat ishin të nevojshme për të instaluar në shfletues në mënyrë që faqet apo aplikacionet që ofronin komunikim të punonin si duhet. Kjo zgjidhje nuk është ideale, sepse ngarkon dhe detyron përdoruesit me mirëmbajtje të panevojshe për të instaluar dhe kontrolluar për azhurnimet e këtyre shtojcave për të cilat disa përdorues mund të mos kenë njohuritë e nevojshme.

Nga fushat kryesore që është revolucionarizuar nga teknologjia kompjuterike është fusha e mjekësisë dhe kujdesit shëndetësor. Në këtë fushë janë sjellë shumë ndryshime duke filluar nga organizimi i të dhënave për pacientët deri te menaxhimi i sallave operative. Shumë vende tani përdorin sisteme për konsultim online mes mjekut dhe pacientit, sisteme këto që mundësojnë të bëhet diagnoza, trajtimi dhe monitorimi i sëmundjes.

Trajtimi mjekësor i bazuar në internet po bëhet shumë i popullarizuar dhe gjithashtu i dobishëm sidomos tani në këto ditë pandemie kur shumë shtete janë në karantinë.

Njerëzit që kanë probleme shëndetësore nuk janë në gjendje të vizitojnë mjekët për shkak të pandemisë, kështu që përdorimi i një softueri që mundëson të komunikojnë me mjekun përkatës do të ishte më se i dobishëm. Vendi ynë është ende mbrapa nga kjo teknologji, por nevoja për këtë lloj softueri për komunikim me mjekun në distancë, tani kur pacientët nuk mund të vizitojnë klinikat është e domosdoshme.

Duke u nisur nga kjo qëllimi jonë do të jetë në gjetjen e një zgjidhje për video komunikim me mjekë nga shtëpia, pra kjo zgjidhje do t'a bëjë të mundëshme konsultimin me mjekun pa pasur nevojë të udhëtohet për në spital, e gjithashtu t'i lejojë mjekët t'i kontaktojnë pacientët e tyre dhe të zhvillojnë biseda lidhur me sëmundjet e tyre.

Në vijim të temës do të flasim në mënyrë të detajuar për funksionet e sistemit dhe zgjidhjet teknike që do t'i bëjmë, ku përveq të tjerash do të fokusohemi në dizajnin dhe në eksperincën e aplikacionit për përdoruesit të cilët mund të mos kenë shumë njohuri me teknologjinë e veçanarisht me aplikacionet.

2 SHQYRTIMI I LITERATURËS

Në këtë kapitull do të analizojmë se çka janë sistemet për komunikim online, rëndësia e përdorimit të sistemeve të tilla për një spital apo klinikë mjekësore.

Gjatë këtij kapitulli gjithashtu do të analizojmë se çka është telemjekësia, komponentët e saj si dhe rëndësia e përdorimit të telemjekësisë dhe cilat janë benefitet bazë që ajo i ofron, pas kësaj do të bëjmë analizimin e sistemeve të ngjashme me sistemin që ne do zhvillojmë, mënyrat se si këto sisteme punojnë dhe si këto sisteme operojnë në treg, e në fund do të bëjmë krahasimin mes tyre.

2.1 Sistemet për komunikim

Një nga sfidat më të mëdha me të cilat përballlet njerëzimi në shekullin 21 është ofrimi i kujdesit shëndetësor për të gjithë njerëzit pa dallim vendndodhje. Ky parashikim është shprehur nga Organizata Botërore e Shëndetësisë (OBSH) në strategjinë e saj shëndetësore për shekullin 21. [1]

Qasja, barazia, cilësia dhe kosto janë çështjet kryesore me të cilat përballlet kujdesi shëndetësor në vendet e zhvilluara dhe ato më pak të zhvilluara ekonomikisht. Sidoqoftë, përparimet e fundit në teknologjinë e informacionit dhe komunikimit (TIK) kanë krijuar shumë mundësi për tejkalimin e këtyre barrierave duke rritur numrin e mënyrave që kujdesi shëndetësor të mund të ofrohet dhe të jetë i çashtë për të gjithë.

Teknologjitë moderne siç janë kompjuterët, celularët dhe interneti, po revolucionarizojnë mënyrën se si individët komunikojnë me njëri-tjetrin, gjithashtu këto teknologji kanë potenciale të mëdha për të ndihmuar në adresimin e problemeve bashkëkohore shëndetësore globale.

Me zhvillimin e internetit ka pasur një zgjerim të madh të komunikimit online. Avantazhet e komunikimit në internet janë në dispozicion të lartë dhe se ky komunikim është kryesisht pa tarifë – përfshirë këtu vetëm koston e lidhjes në internet.

Në vitin 2018 rreth 76,62 përqind e popullsisë së Republikës së Kosovës ka qasje në internet. [2]

Kjo përqindje e lartë e disponueshmërisë e bën internetin platformë të shkëlqyeshme për komunikim të përditshëm me familjen, miqtë, kolegët etj.

Komunikimi në internet mund të ndahet bazuar në llojin e mesazheve të transferuara. Forma më e përdorur gjerësisht e komunikimit në ditët e sotme është komunikimi me tekst. Shembuj të këtij komunikimi janë email dhe rrjetet sociale. Tjetër formë e komunikimit majft e përhapur në ditët e sotme është edhe komunikimi multimedial. Ky komunikim transferon audio, video ose kombinim e të dyjave në kohë reale. Avantazhi kryesor i këtij komunikimi është mundësia e reagimit të menjëhershëm sikurse në komunikimin personal në jetë reale.

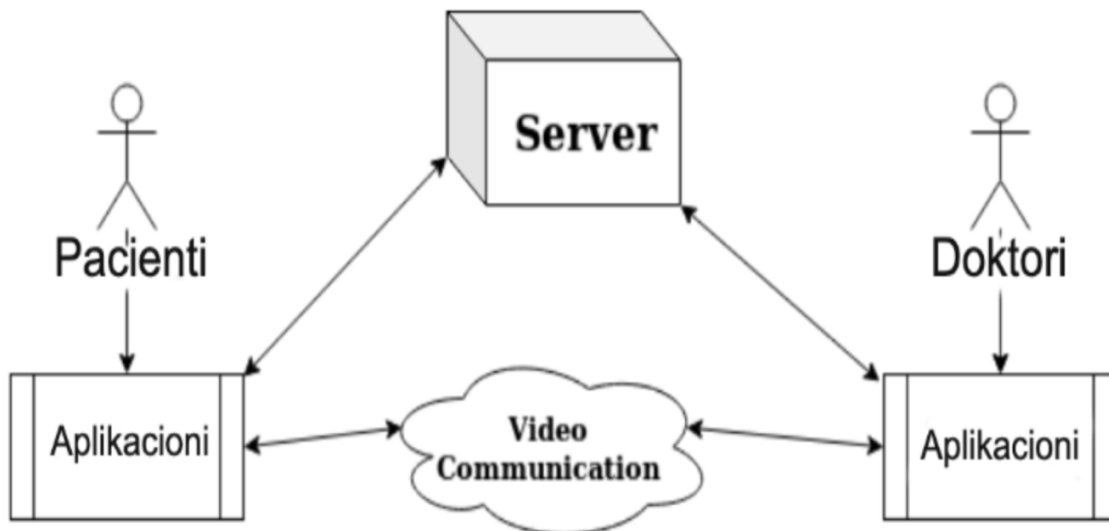


Figura 1. Topologjia e video komunikimit

2.2 Telemjekësia dhe komponentët e saj

Telemjekësia është ofrimi i shërbimit mjekësor nga distanca, përmes përdorimit të teknologjisë së informacionit dhe komunikimit elektronik që mundëson shkëmbimin e informacioneve shëndetësore të dobishme për diagnozën, trajnimin dhe parandalimin e sëmundjeve dhe lëndimeve. [3]

Telemjekësia përdor teknologji të tilla si kompjuter ose video për të ofruar shërbimet e kujdesit shëndetësor. Ofruesit e kujdesit shëndetësor mund të përdorin telemjekësinë për të mësuar lidhur me pacientët dhe t'i trajtojnë ata në distancë. Kështu, nëse pacienti jeton larg ofruesit të shërbimit shëndetësor ose nuk arrin të shkojë në një qendër mjeksore kur i nevojitet, telemjekësia vjen në ndihmë.

Telemjekësia nuk është specialitet mjekësor i veçantë. Shumë specialistë, si psh.: doktorë alergjish, dermatologë etj., mund të kujdesen për pacientët e tyre përmes telemjekësisë. Studiuesit e shëndetit mund t'a përdorin telemjekësinë gjithashtu edhe për të mbledhur informacione nga vullnetarët në studim ("pjesëmarrësit në studim" ose "subjektet studimore").

Tri kategoritë apo modulet kryesore të telemjekësisë janë:

Telekonsulta: Kjo është një nga kategoritë më të përdorura të telemjekësisë. Në thelb i referohet komunikimit në internet ku zhvillohet një konsultë midis mjekut dhe pacientit. Mund të bëhet përmes përdorimit të teknologjisë së videokonferencës ose thjesht duke përcjellë një fotografi ose informatë të veçantë ofruesit të kujdesit shëndetësor për interpretime të mëtejshme. [4]

Telementorimi: Kjo kategori ka të bëjë me situatat kur një mjek merr mentorim nga një mjek tjetër, i cili mund të jetë më i specializuar për të gjetur mënyra për të ndihmuar një pacient. Për shembull një mjek i cili kërkon ndihmë në menaxhimin e një pacienti të veçantë, gjëja më e mirë për të bërë është që të lidhet me një institucion shëndetësor që do t'i sigurojë një mentor i cili do t'i ofron ndihmë përmes sistemit, e gjithë kjo procedurë është telementorim. [4]

Telemonitorimi: Ka të bëjë me rastet kur një mjek merr të dhëna nga një pacient në shtëpi përderisa ai është i lidhur me një sensor të caktuar që mund të monitorojë në mënyrë efektive gjendjen e tij. Mjeku më pas do të përdorë rezultatet që ata kanë marrë nga sensorët në sistem për të filluar administrimin e trajtimit ose përshkrimin e ilaçeve për pacientin. Kjo formë e telemjekësisë lejon që një pacient të monitorohet lehtësisht nga shtëpia, pa pasur nevojë të udhëtojë në institucionin shëndetësor. [4]

2.3 Analizimi i Sistemeve për komunikim

Sistemet digjitale për komunikim janë duke u bërë ndër sistemet kryesore të komunikimit, njëra ndër arsyet kryesore të popullarizimit të këtyre sistemeve padyshim është kosto e ulët. Këto sisteme përdoren për arsyet e ndryshme duke filluar nga rekreacioni e deri në mjekësi. Në këtë pjesë ne do të bëjmë analizimin e tri sistemeve që janë më të popullarizuara në botë për telemjekësi dhe do të mundohemi t'i krahasojmë ato.

2.3.1 Doxy.me

Doxy.me është ndër sistemet më të famshme për komunikim me pacientë, i cili ofron mundësinë e komunikimit me video dhe zë, ruan historinë e thirrjeve si dhe mundëson transferimin e dokumenteve të ndryshme. [5]

Ky sistem, njëjtë sikur sistemi që do e zhvillojmë ne, gjithashtu është i zhvilluar duke përdorur teknologjinë webRTC.

Sistemi ka dashboard e cila ka pamje të tillë si në figurën më poshtë:

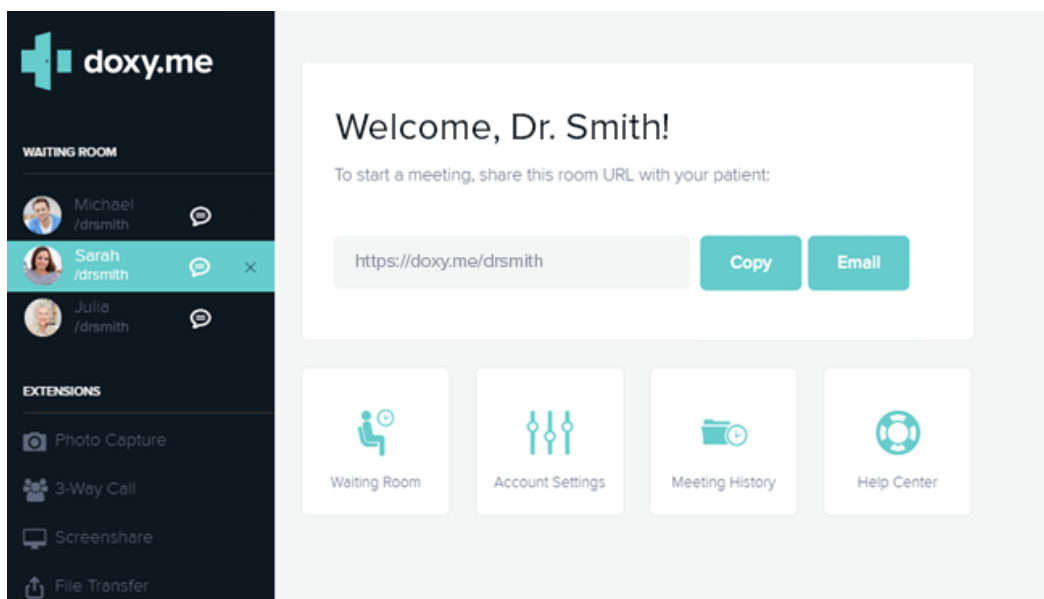


Figura 2. Sistemi Doxy.me

Për të përdorur këtë sistem fillimisht secili përdorues duhet të regjistrohet dhe pastaj ai pajiset me email dhe fjalëkalim me të cilat mund të kyçet në sistem.

Forma për t'u kyçur në sistem duket si në figurën më poshtë :

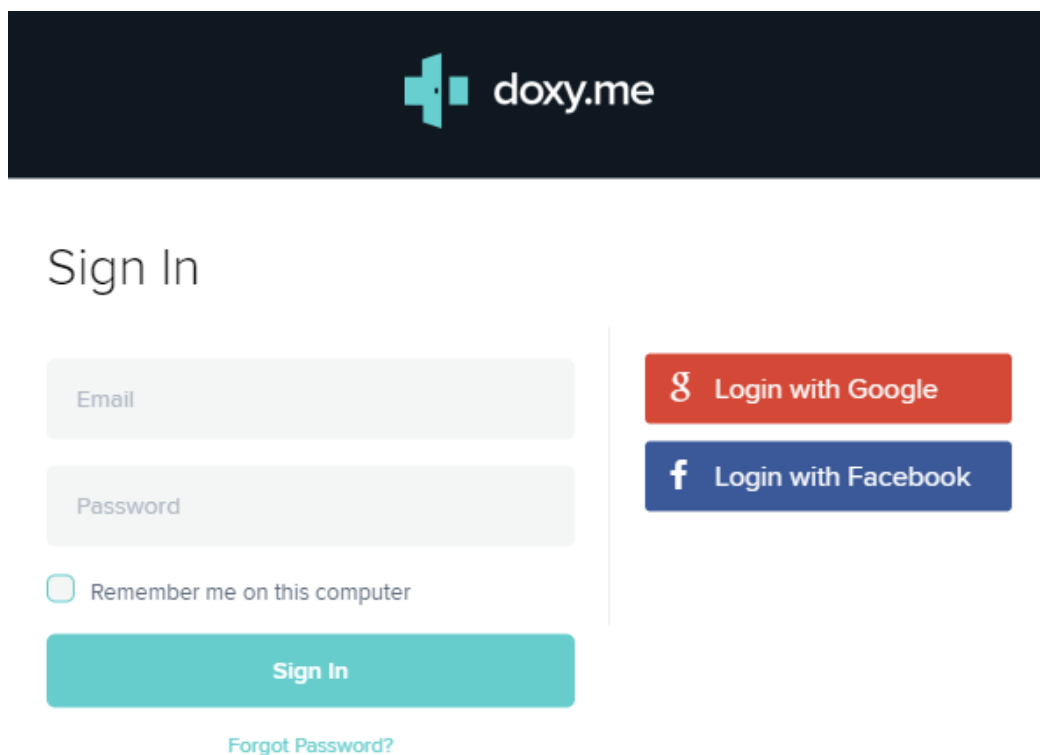


Figura 3. Doxy.me Login

Doxy.me është ndërtuar dhe operon në treg si “Software as a Service”, ky sistem është i programuar në gjuhën Javascript (Node.js) dhe përkrah shumëgjuhësinë.

Ky sistem është në dispozicion vetëm nëpërmjet browserit (shfletuesit), pra nuk ka një mobile app për telefonët Android dhe për produktet e Apple si iPad dhe iPhone.

Doxy.me e ofron mundësinë e përdorimit të këtij sistemi pa pagesë për një kohë të limituar prej 30 ditësh, e cila për t'u vazhduar duhet të paguhet. [5]

2.3.2 eVisit

eVisit është ndër sistemet e para të këtij lloji në treg për komunikim me pacientë, i cili ofron mundësinë e komunikimit me video dhe zë, transferimin e dokumenteve të ndryshme si dhe monitorimin e gjendjes së pacientit. [6]

Dashboardi i këtij sistemi duket si më poshtë:

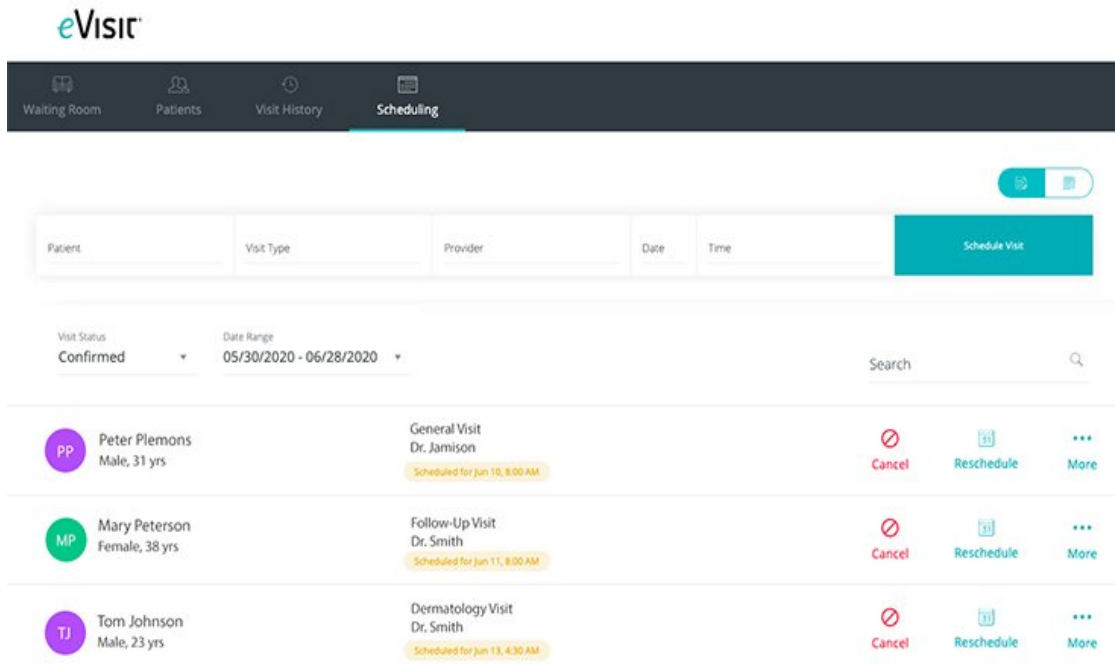


Figura 4. Sistemi eVisit

Secili përdorues që dëshiron t'a përdorur këtë sistem fillimisht duhet të regjistrohet e pastaj pajiset me email dhe një fjalëkalim me të cilat mund kyçet në sistem. Forma për qasje në sistem duket si në figurën më poshtë:

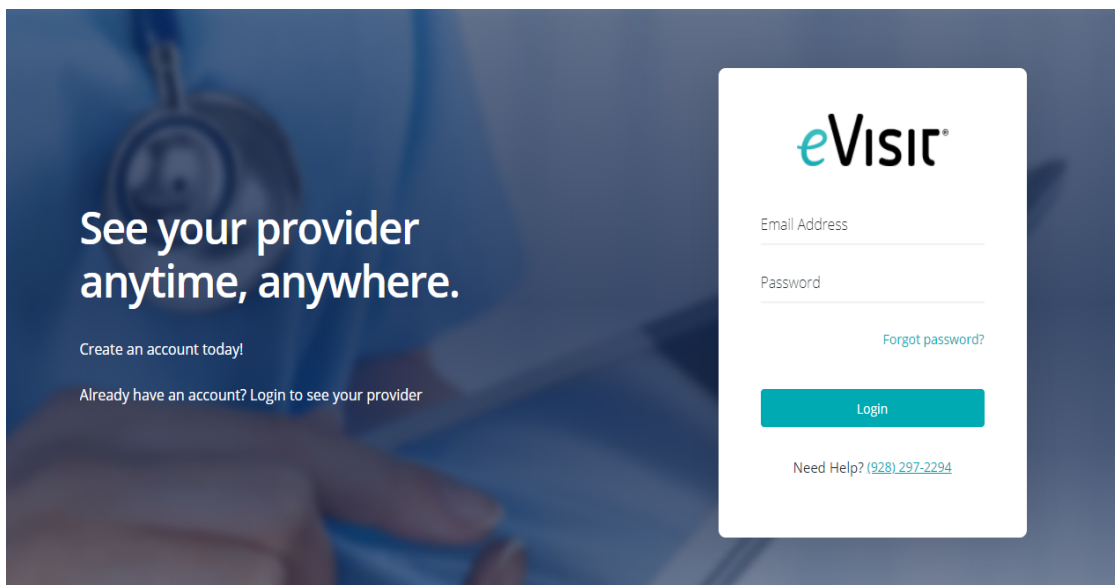


Figura 5. eVisit Login

Edhe ky sistem njësoj sikurse Doxy.me është zhvilluar duke përdorur teknologjitë për komunikim në kohë reale WebRTC, RTP dhe SIP, si dhe gjuhët programuese Node.js dhe Python.

eVisit nuk ofron mobile app për paisjet mobile android edhe ios, e as për ato desktop, pra është në dispozicion vetëm në platformen web.

Ky sistem e ofron mundësinë e përdorimit të trial (pa pagesë) për një kohë shumë të limituar vetëm si demo, e cila për t'u vazhduar duhet të paguhet. [6]

2.3.3 Teladoc

Teladoc është sistem i cili ju mundëson pacientëve dhe mjekëve komunikimin me video dhe zë dhe është ndër sistemet më të kompletuara të këtij lloji. Përveq web platformes ky sistem ekziton edhe ne platformen mobile, ku edhe e ka fokusin kryesor të zhvillimit. [7]

Dashboardi i këtij sistemi duket si më poshtë:

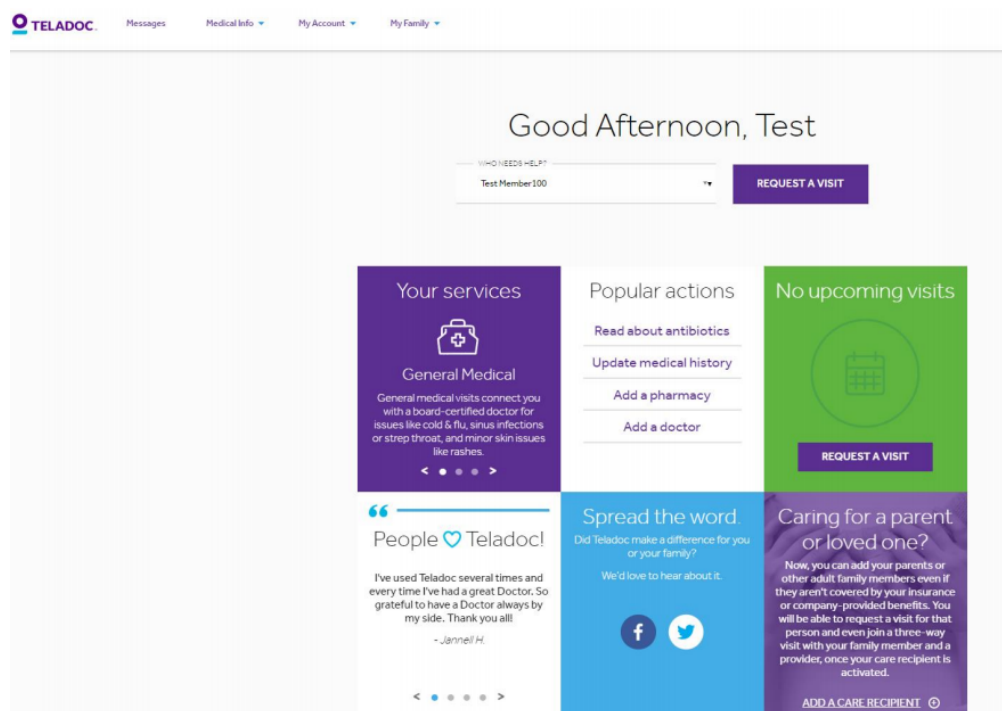


Figura 6. Sistemi Teladoc

Për të përdorur këtë sistem fillimisht secili përdorues duhet të plotësoj formën e regjistrimit e pastaj të pajiset me email dhe një fjalëkalim me të cilat mund kyçet në sistem.

Forma për qasje në sistem duket si në figurën më poshtë:

Figura 7. Teladoc Login

Edhe ky sistem gjithashtu është zhvilluar duke përdorur teknologjinë për komunikim në kohë reale WebRTC. Për Mobile App është në dispozicion për telefonët Android dhe për produktet e Apple si iPad dhe iPhone. Për përdoruesit e këtij sistemi varësisht se cilin edicion e përdorin ky sistem ofron hapsirë të limituar për ruajtjen e të dhënave.

Teladoc nuk e ofron mundësinë e përdorimit të këtij sistemi pa pagesë. [7]

2.3.4 Krahasimi i sistemeve

Pas analizimit të sistemeve më lartë, në tabelën me numër 1 do t'i krahasojmë sistemet më të përdorura në botë për telemjekësi. Në tabelën 1 do të krahasojmë funksionet e tre sistemeve që janë më të përdorura për nga karakteristikat që ato i ofrojnë.

Funkisonet	Doxy.me	eVisit	Teladoc
Gjuha e zhvillimit	Javascript	Python	Javascript

webRTC	Po	Po	Po
Mobile App	Jo	Jo	Po
Çmimi/Muaj	29 \$	600 \$	49 \$

Tabela 1. Krahasimi i sistemeve

Pas krahasimeve në bazë të funksioneve dhe çmimeve del që sistemi eVisit është sistemi me çmimin më të lartë prej 600\$/muaj në krahasim me Doxy.me që ka çmimin 29\$ dhe Teladoc që kushton 49\$.

Sa i përket aplikacioneve mobile pas krahasimit del që vetëm Teladoc ofron zgjidhje per platformen mobile android dhe ios. Poashtu shohim që të tri këto sisteme përdorin teknologjinë për komunikim në kohë reale webRTC.

3 DEKLARIMI I PROBLEMIT

Si në çdo sferë të jetës zhvillimi i teknologjisë ka ndikuar edhe në mjekësi, ku janë zhvilluar teknologji dhe aparatura të ndryshme. Zhvillim të madh ka pasur edhe telemjekësia si pjesë e mjekësisë ku janë zhvilluar sisteme të ndryshme të kësaj lëmie.

Këto sisteme mund të i'u sjellin shumë benefite pacientëve dhe spitaleve për konsultë, diagnostifikim dhe monitorim të sëmundjeve të ndryshme, e që mund të sjellin edhe deri tek shpëtimi i jetës.

Sistemet ekzistuese dhe ato që i përmendëm më lartë kanë edhe mangësi të ndryshme, disa nga mangësitë që vlenë të theksohen janë mos përkrahja e shumicës së shfoltuesve dhe mospërkrahja e gjuhës sonë amtare, e të cilat kanë rëndësi të madhe për ne.

Duke u nisur nga nevoja dhe mundësia e zhvillimit të një sistemi të tillë, një spital në vendin tonë ka ofruar përkrahjen, që pas zhvillimit versionin beta të sistemit të mund t'a testojmë në institucionin e tyre, gjë që do të lehtësojë shumë pjesën e implementimit dhe testimit në mënyrë të suksesshme.

4 METODOLGJIA

Për t'u informuar dhe kuptuar më shumë në lidhje me temën dhe me sisteme të ngjajshme dhe teknologjitë e tyre, ne kemi bërë hulumtime në artikuj shkencorë dhe materiale të ndryshme, në metodologjinë tonë kemi përfshirë disa metoda siç janë:

Përmes shfletimit të literaturës kemi analizuar sisteme të ngjajshme që përdoren në botë, teknologjinë që këto sisteme përdorin dhe kemi kuptuar mendimin e zhvilluesve të këtyre sistemeve.

Analizimi i të dhënave – Kemi hulumtuar materiale të ndryshme rreth funksionimit të sistemeve tjera të ngjashme me sistemin që ne do e zhvillojmë, kemi analizuar gjithashtu se si këto sisteme u sjellin benefite dhe ju ndihmojnë spitaleve dhe klinikave të ndryshme në telemjekësi, pra komunikimin, diagnostifikimin dhe monitorimin e pacientëve në distancë.

Krahasimi i sistemeve – metodë kjo që është përdorur për të kuptuar përparsitë dhe mangësitë e sistemeve që kemi analizuar më lartë.

Zhvillimi i sistemit – Pasi analizimit të sistemeve të ngjajshme dhe funksioneve më të përdorura të këtyre sistemeve, ne kemi ardhur në një përfundim dhe jemi munduar të zhvillojmë një sistem, i cili i përshtatet dhe i përmbushë kërkesat e një spitali në tregun tonë

5 REZULTATET

Pas analizimit të sistemeve që kanë të bëjnë me komunikim online në telemjekësi e kemi parë të arsyeshme nevojën për zhvillimin e një sistemi të tillë për një spital në tregun tonë. Ideja për dizajnimin dhe zhvillimin e këtij aplikacioni për komunikim online mes pacientit dhe doktorit përmes sistemit tonë ka ardhur pas shumë arsyeve dhe kërkesave që kemi parë, duke filluar nga gjendja e rëndë pandemike në të cilën gjendemi momentalisht, e që është shumë e rëndësishme mbajtja e distancës dhe qendrimi larg vendeve me shumë njerëz në menyrë që t'i shmangemi kontakteve.

Me lehtësinë që e ofron teknologjia webRTC e cila është shumë e avansuar, aplikacioni ynë ndihmon sado pak në përmirësimin dhe lehtësimin e këtyre rasteve.

Ky sistem do të zhvillohet në platformën web përmes përdorimit të teknologjive të ndryshme si: PHP, HTML, CSS3, Javascript, jQuery dhe për bazë të të dhënave do të përdorim MySQL.

Ky sistem do të ofrojë module të ndryshme siç janë:

- Staf i mjeksorë,
- Pacientët,
- Departamentet dhe
- Doktorët

Kto module do të zhvillohen në bazë të kërkesave të cilat i kemi shtjelluar dhe do të falsim më vonë.

Në vijim do t'i përshkruajmë shkurtimisht se çka përmbajnë modulet e aplikacionit. Staf i mjeksorë është moduli i parë, ky modul ka autorizime të administratorit, ku stafit i ofrohet mundësia të shohë listën e pacientave, doktorëve, termineve, departamenteve, si dhe regjistrimin apo ndryshimin e tyre. Në modulën e dytë bëhet fjalë për pacientët të cilët kanë mundësinë të caktojnë termine tek doktori specifik ku pastaj mund të zhvillojnë video-bisedën, në modulën e tretë gjenden departamentet apo repartet ku secili departament ka doktorët e tij, në këtë modul mund të shtohet apo të ndryshohet një departament, ndërsa në

modulin e fundit bëhet fjalë për doktorët të cilët kanë mundësinë të shohin terminet e tyre, ku pastaj mund edhe të zhvillojnë video-bisedën me pacientët.

5.1 Përshkrimi i moduleve të sistemit

Që nga fillimi qëllimi ynë ishte që ky sistem të jetë shumë i thjeshtë dhe të ketë një ndërfaqe të lehtë për përdoruesin, gjithashtu të jetë kompatibil për të gjithë ueb shfletuesit.

Në këtë pjesë ne do të përshkruajmë modulet të cilat janë zhvilluar në sistem hap pas hapi.

5.1.1 Kycja në sistem

Së pari, çdo lloj përdoruesi, si stafi mjekësor, doktorë apo pacientët mund të identifikohen në këtë sistem duke siguruar emrin e përdoruesit dhe fjalëkalimin. Kjo është faqja fillestare e sistemit tonë që kemi zhvilluar. Kur ndonjë përdorues viziton sistemin, ai do të kërkojë kredencialet për të përdorur këtë sistem.

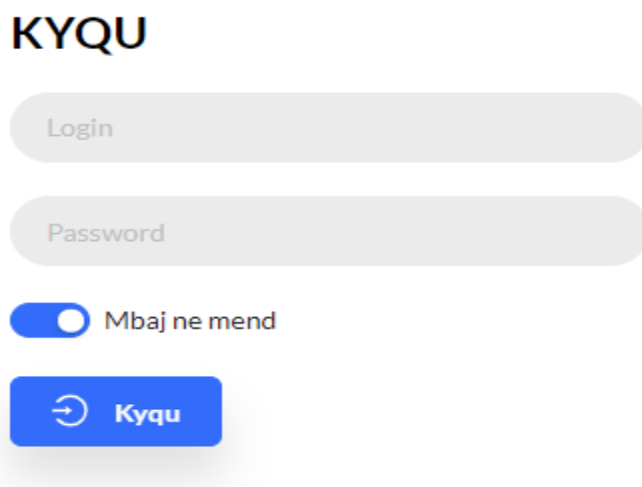


Figura 8. Login për përdoruesin

Për kyqje si dhe për validim se a ekziston ky përdorues në sistem është përdorur kjo pjesë e kodit më poshtë:

```
public function login(){
    $data = array();
    if($this->session->userdata('success_msg')){
        $data['success_msg'] = $this->session->userdata('success_msg');
        $this->session->unset_userdata('success_msg');
    }
    if($this->session->userdata('error_msg')){
        $data['error_msg'] = $this->session->userdata('error_msg');
        $this->session->unset_userdata('error_msg');
    }
    if($this->input->post('loginSubmit')){
        $this->form_validation->set_rules('email', 'Email', 'required|valid_email');
        $this->form_validation->set_rules('password', 'password', 'required');
        if($this->form_validation->run() == true){
            $scon = array(
                'returnType' => 'single',
                'conditions' => array(
                    'email'=> $this->input->post('email'),
                    'password' => md5($this->input->post('password')),
                    'status' => 1
                )
            );
            $checkLogin = $this->user->getRows($scon);
            if($checkLogin){
                $this->session->set_userdata('isUserLoggedIn', TRUE);
                $this->session->set_userdata('userId', $checkLogin['id']);
                redirect('users/account/');
            }
        }
    }
}
```

```

    }else{
        $data['error_msg'] = 'Email ose Password gabim, ju lutem provoni perseri!';
    }
}else{
    $data['error_msg'] = 'Ju lutem plotesoni te gjitha fushat e kerkuara!';
}
}
$this->load->view('elements/header', $data);
$this->load->view('users/login', $data);
$this->load->view('elements/footer');
}

```

5.1.2 Moduli i adminit për stafin mjeksorë

Në këtë modul të sistemit, stafi mjeksorë mund të veprojnë si administratorë të sistemit. Këtu ata kanë të integruar ndërfaqen e plotë të personalizuar ku mund të menaxhojnë të gjithë sistemin e brendshëm, disa prej këtyre funksioneve nuk janë të arritshëm nga paneli i doktorëve dhe ai i pacientëve.

5.1.2.1 Dashboard

Dashboard-i shfaq përmbledhje të statistikave të sistemit. Stafii mjeksorë mund të shoh numrin e mjekëve të regjistruar, numrin e pacientëve të regjistruar, të dhënat për çdo termin të pacientit apo doktorit.

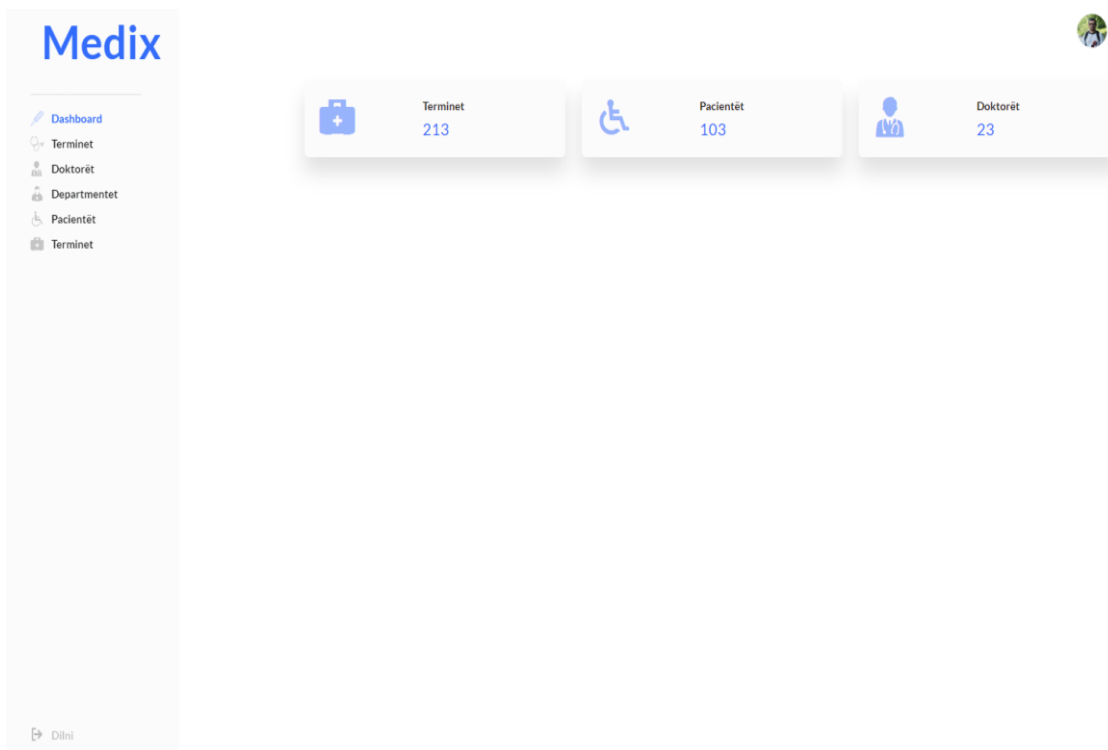


Figura 9. Dashboard për stafin mjeksorë

Kodi i përdorur për dashboard:

```
public function dashboard()
{
    $this->load->model('Statistics');
    $data['appointments'] = $this->Statistics->count_appointments();
    $data['patients'] = $this->Statistics->count_patients();
    $data['doctors'] = $this->Statistics->count_doctors();
    //load data into dashboard view
    $this->load->view('dashboard', $data);
}
```

5.1.2.2 Lista e pacientëve

Nga menyuja e majtë, duke klikuar në opsionin "Pacientët", stafi mjekësorë mund të shoh të gjithë listën e informacionit për pacientët që janë të regjistruar në sistem. Lista tregon emrin, id e pacientit, moshën, gjinin dhe numri e kontaktit.

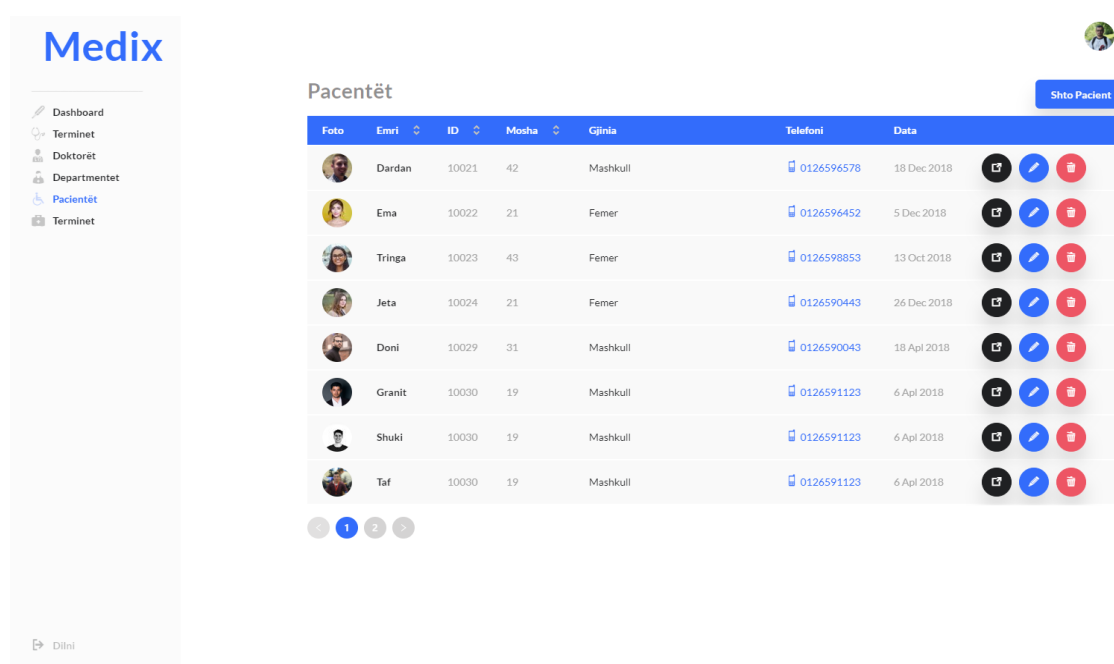


Foto	Emri	ID	Moshë	Gjinia	Telefoni	Data			
	Dardan	10021	42	Mashkull	0126596578	18 Dec 2018			
	Ema	10022	21	Femer	0126596452	5 Dec 2018			
	Tringa	10023	43	Femer	0126598853	13 Oct 2018			
	Jeta	10024	21	Femer	0126590443	26 Dec 2018			
	Doni	10029	31	Mashkull	0126590043	18 Apl 2018			
	Granit	10030	19	Mashkull	0126591123	6 Apl 2018			
	Shuki	10030	19	Mashkull	0126591123	6 Apl 2018			
	Taf	10030	19	Mashkull	0126591123	6 Apl 2018			

Figura 10. Lista e pacientëve për stafin mjekësorë

Kodi i përdorur për shfaqjen e pacientëve:

```
public function patients()
{
    $this->load ->model('Patients');
    $active_patients = TRUE;
    $data[patients] = $this->'Patients' ->get_patients($active_patients);
    $this->load->view('patients', $data);
}
```

5.1.2.3 Shtimi i një pacienti

Stafi mjekësor mund të shtojë pacient të ri duke klikuar në butonin “Shto Pacient” nga faqja e listës së pacientëve. Pacienti shtohet pasi të mbushën fushat me të dhënat të cilat i kërkon forma për shtimin e pacientit, sic janë: username, fjalëkalimin, emrin, moshën, gjininë, numrin e kontaktit, foton, etj. Duke klikuar në butonin ‘Ruaj’, stafi mjekësorë mund të shtojë pacientin e ri.

The screenshot shows the 'Shto Pacientin' form in the Medix application. The form is titled 'Shto Pacientin' and includes a profile picture upload button labeled 'Zgjedh foton'. The form fields are as follows:

- Emri:** First name field containing 'Fitim'.
- Mbiemri:** Surname field containing 'Mehmeti'.
- Email adresa:** Email address field containing 'mehmetfitim92@gmail.com'.
- Gjinia:** Gender dropdown menu set to 'Maskull'.
- Adresa:** Address field containing 'Shipol Mitrovicë'.
- Telefoni:** Phone number field containing '049132654'.
- Moshë:** Age field containing '20'.

A blue 'RUAJ' button is located at the bottom of the form.

Figura 11. Faqja për shtimin e një pacienti

Kodi i përdorur për shtimin e pacientëve:

```
public function add_patient(): void
{
    $this->load ->library('form_validation');
    $this->form_validation->set_rules('name', 'Emri', 'required');
    $this->form_validation->set_rules('surname', 'Mbiemri', 'required');
    $this->form_validation->set_rules('email','Email ',
                                     'required|valid_email|is_unique[users.email]');
```

```

$this->load->model('Patient');
if ($this->form_validation->run() == FALSE)
{
    $data['success'] = FALSE;
    $this->load->view('patient', $data);
    return;
}
$data['patient'] = [
    'name' => $this->input->post('name'),
    'surname' => $this->input->post('surname'),
    'email' => $this->input->post('email'),
    'gender' => $this->input->post('gender'),
    'address' => $this->input->post('address'),
    'phone' => $this->input->post('phone'),
    'age' => $this->input->post('age'),
    'role' => 3
];
if($this->Users->add_user($data['patient']))
{
    $data['success'] = TRUE;
}else{
    $data['success'] = FALSE;
}
$this->load->view('patient', $data);
}

```

5.1.2.4 Të dhënat e pacientit

Nga faqja e listës së pacientëve, stafi mjekësor mund të klikojë në butonin për detaje për të parë të gjithë informatat e një pacienti të caktuar. Në këtë faqe stafi mjekësorë mund të shoh emrin, foton, moshën, gjininë, kontaktin, etj, dhe mund t'i ndryshojë këto të dhëna duke klikuar “Ruaj”.

The screenshot shows the 'Medix' patient profile page. On the left is a sidebar with navigation links: Dashboard, Terminet, Doktorët, Departamentet, Pacientët, and Terminet. The main content area is titled 'Profili i Pacientit' and features a profile picture placeholder with a 'Zgjedh foton' button. Below this are form fields for: Emri (First name: Fitim), Mbiemri (Last name: Mehmet), Email adresa (mehmetfitim92@gmail.com), Gjinia (Mashkull), Adresa (Shijol Mitrovice), Telefoni (0491322654), and Mosha (20). A blue 'RUAJ' button is at the bottom.

Figura 12. Faqja për të dhënat e pacientit

Kodi i përdorur për shfaqjen dhe editimin e pacientit:

```
public function patient(int $patient_id): Patient
{
    // if no $patient_id is assigned. Return an empty object!
    if(!$patient_id)
    {
        return new Patient;
    }
    $this->load->model('Patient');
    $data['patient'] = $this->Patients->get_patient($patient_id);
    $this->load->view('patient', $data);
}
```

5.1.2.5 Lista e doktorëve

Nga menyuja e majtë, duke klikuar në opsionin "Doktorët", stafi mjeksorë mund të shohë të gjithë listën e informacionit për doktorët që janë të regjistruar në sistem.

Lista tregon emrin, moshën, gjininë, numri i kontaktit dhe departamenti të cilit i takon ai doktor.

Foto	Emri	ID	Moshë	Gjinia	Telefoni	Departamenti	
	Festim	10021	42	Mashkull	0111196578	Neurologji	
	Lina	10022	21	Femer	0122336452	Kirurgji	
	Dita	10024	21	Femer	0998520443	Kardiologji	
	Armend	10029	31	Mashkull	0112266903	Neurologji	
	Dardan	10030	19	Mashkull	0126591123	Neurologji	
	Fatlum	10030	19	Mashkull	0003911203	Gastroenterologji	
	Faton	10030	19	Mashkull	3625591123	Gastroenterologji	

Figura 13. Lista e doktorëve

Kodi i përdorur për shfaqjen e doktorëve:

```
public function doctors()
{
    $this->load->model('Doctors');
    $active_doctors = TRUE;
    $data[doctors] = $this->'Doctors'->get_doctors($active_doctors);
    $this->load->view('doctors', $data);
}
```


5.1.2.6 Shtimi i një doktori

Stafi mjekësorë mund të shtojë doktor të ri duke klikuar në butonin “Shto Doktor” nga faqja e listës së doktorëve. Doktor shtohet pasi të mbushen fushat me të dhënat të cilat i kërkon forma për shtimin e doktorit, sic janë: username, fjalëkalimin, emrin, moshën, gjininë, numrin e kontaktit, foton, adresën, departamentin të cilit i takon etj, dhe klikohet butoni “Ruaj”.

The screenshot shows the 'Shto Doktorin' form in the Medix application. The form has the following fields and values:

- Emri:** Festim
- Mbiemri:** Mehmeti
- Departamenti:** Kirurgji
- Gjinia:** Mashkull
- Adresa:** Dëshmorët e Kombit, Mitrovicë
- Telefoni:** 049111222
- Moshë:** 39

A blue 'RUAJ' button is located at the bottom of the form. A sidebar on the left contains navigation links: Dashboard, Terminet, Doktorët, Departamentet, Pacientët, and Terminet.

Figura 14. Faqja për shtimin e një doktori

Kodi i përdorur për shtimin e doktorit:

```
public function add_doctor(): void
{
    $this->load->library('form_validation');
    $this->form_validation->set_rules('name', 'Emri', 'required');
    $this->form_validation->set_rules('surname', 'Mbiemri', 'required');
    $this->form_validation->set_rules('email', 'Email',
        'required|valid_email|is_unique[users.email]');
    $this->load->model('Doctor');
    if ($this->form_validation->run() == FALSE)
    {
```

```

$data['success'] = FALSE;
$this->load->view('doctor', $data);
return;
}
$data['doctor'] = [
    'name' => $this->input->post('name'),
    'surname' => $this->input->post('surname'),
    'email' => $this->input->post('email'),
    'gender' => $this->input->post('gender'),
    'department' => $this->input->post('department'),
    'address' => $this->input->post('address'),
    'phone' => $this->input->post('phone'),
    'age' => $this->input->post('age'),
    'role' => 3
];
if($this->Users->add_user($data['doctor']))
{
    $data['success'] = TRUE;
} else {
    $data['success'] = FALSE;
}
$this->load->view('doctor', $data);
}

```

5.1.2.7 Të dhënat e doktorit

Nga faqja e listës së doktorëve, stafi mjekësor mund të klikojë në butonin për detaje për të parë të gjithë informatat e një doktorit të caktuar. Në këtë faqe stafi mjekësor mund të shohë emrin, foton e doktorit, moshën, gjininë, kontaktin, departamentin të cilit ai doktor i takon etj, dhe mund t'i ndryshojë këto të dhëna duke klikuar “Ruaj”.

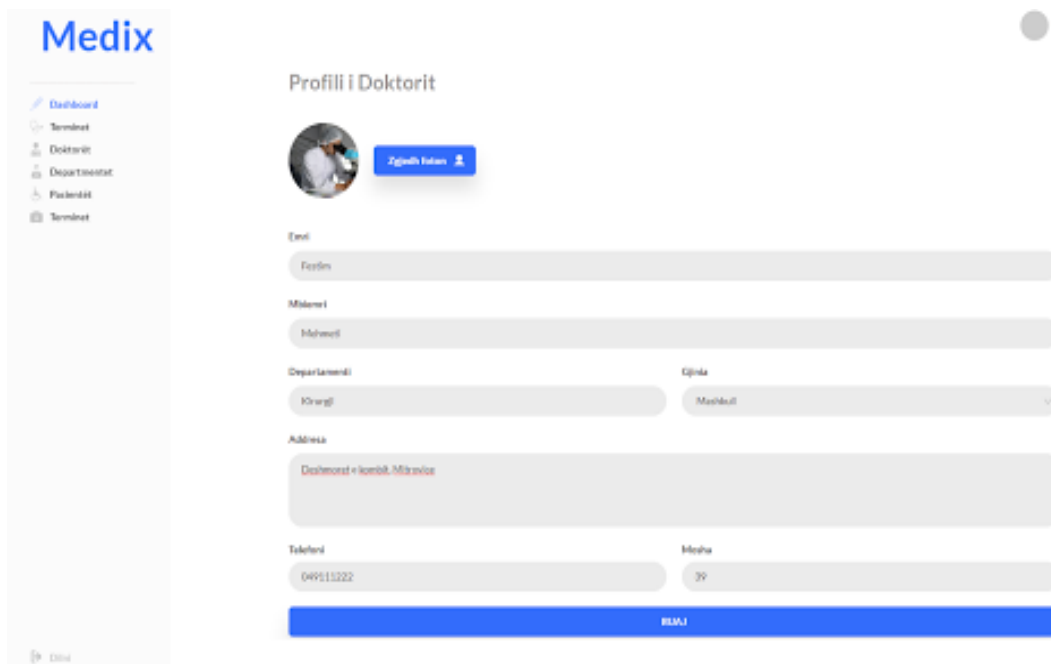


Figura 15. Faqja për të dhënat e doktorit

Kodi i përdorur për shfaqjen dhe editimin e doktorit:

```
public function doctor(int $doctor_id): Doctor
{
    // if no $doctor_id is assigned. Return an empty object!
    if(!$doctor_id)
    {
        return new Doctor;
    }
    $this->load->model('Doctor');
    $data['doctor'] = $this->Doctors->get_doctor($doctor_id);
    $this->load->view('doctor', $data);
}
```

5.1.2.8 Lista e departamenteve

Nga menyuja e majtë, duke klikuar në opsionin "Departamentet", stafi mjekësor mund të shohë të gjithë listën e informacionit për repartet në dispozicion që janë të regjistruar në sistem.

Lista tregon emrin e secilit repart, numrin e doktorëve si dhe numrin e doktorëve të atij reparti që gjendet në sistem.

Emri i repartit	Numri i doktorve	Drojtor
Neurologji	12	Dr. Agim Hasi
Kirurgji	8	Dr. Arjeta Gashi
Dermatologji	22	Dr. Artan Krasniqi
Smotatologji	9	Dr. Jeta Shala
Gastroentologji	18	Dr. Mirjeta Morina
ORL	10	Dr. Festim Mehmeti
Oftalmologji	5	Dr. Granit Peja
Psikiatri	15	Dr. Dardan Arifa
Psikologji	13	Dr. Fatlum Mustafa

Figura 16. Lista e departamenteve

Kodi i përdorur për shfaqjen reparteve:

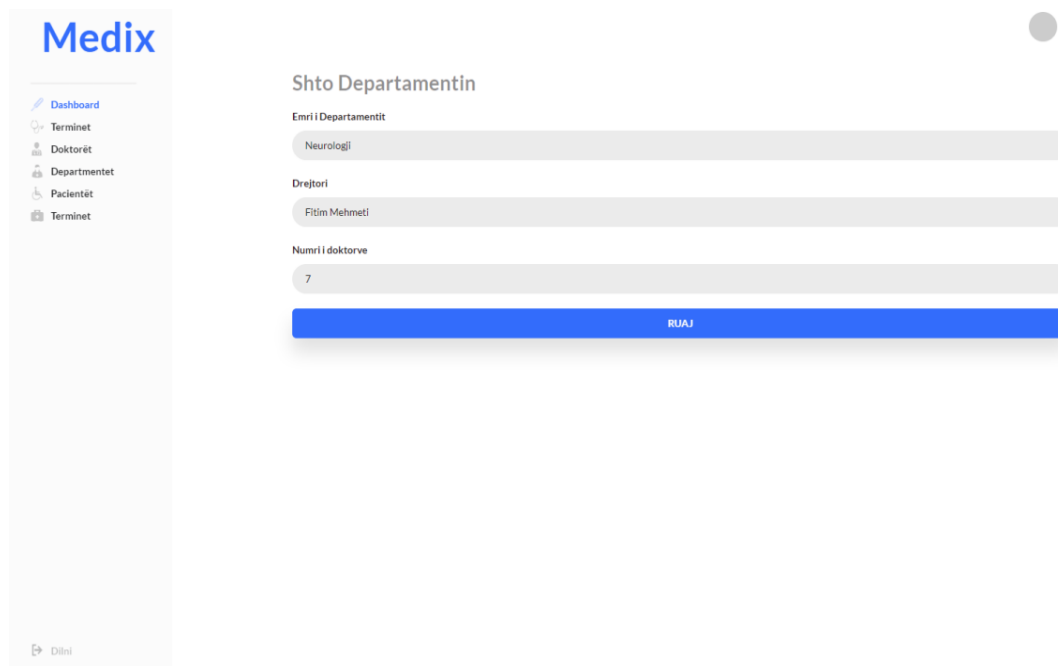
```
public function appointments()
{
    $this ->load ->model('Appointments');
    $active_appointments = TRUE;
    $data[appointments] = $this ->Appointments ->
        get_appointments($active_appointments);
    $this ->load ->view('appointments', $data);
}
```

5.1.2.9 Shtimi i një departamenti

Stafi mjekësor mund të shtojë departament të ri duke klikuar në butonin “Shto Departamentin” nga faqja e listës së departamenteve.

Departamenti mund të shtohet pasi të mbushet fusha me të dhënat të cilat i kërkon forma për shtimin e departamentit në këtë rast është emri i repartit, numri i doktorëve për atë departament, drejtori i atij departamenti etj.

Duke klikuar në butonin ‘Ruaj’, stafi mjekësor mund të shtojë departamenti i ri.



The screenshot shows a web interface for adding a department. On the left is a sidebar with the 'Medix' logo and navigation links: Dashboard, Terminet, Doktorët, Departamentet, Pacientët, and Terminet. The main content area is titled 'Shto Departamentin' and contains three input fields: 'Emri i Departamentit' with the value 'Neurologji', 'Drejtori' with the value 'Fitim Mehmeti', and 'Numri i doktorve' with the value '7'. A blue button labeled 'RUAJ' is positioned below the fields.

Figura 17. Faqja për shtimin e departamentit

Kodi i përdorur për shtimin e reparteve:

```
public function add_department(): void
{
    $this->load->library('form_validation');
    $this->form_validation ->set_rules('name', 'Emri i departamentit ', 'required');
    $this->form_validation ->set_rules('doctor', 'Drejtori i departamentit', 'required');
    $this->form_validation ->set_rules('doctors_nr', 'Numri i doktorve ' 'required');
    $this->load->model('Department');

    if ($this-> form_validation->run() == FALSE)
    {
        $data['success'] = FALSE;
        $this->load -> view('department', $data);
    }
}
```

```

return;
}

$data['department'] = [
    'name' => $this->input->post('department_name'),
    'doctor' => $this->input->post('doctor'),
    'doctors_nr' => $this->input->post('doctors_nr')
];

if($this->Department->add_department($data['department']))
{
    $data['success'] = TRUE;
}else{
    $data['success'] = FALSE;
}
$this->load->view('department', $data);
}

```

5.1.3 Moduli i pacientit

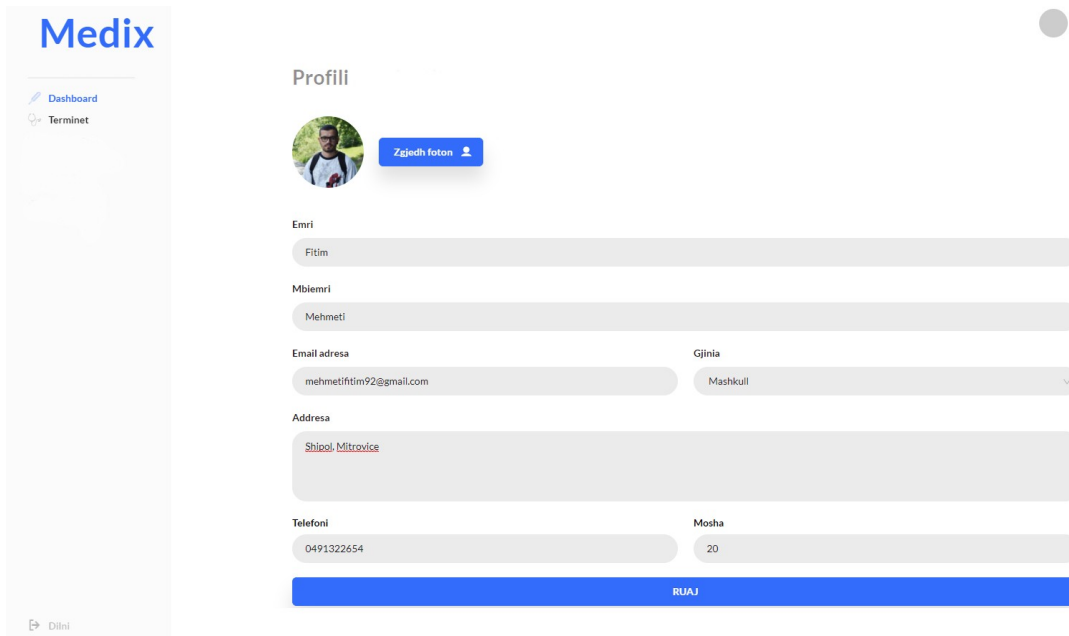
Pasi të jetë krijuar profili i pacientit nga stafi mjekësorë, pacienti mund të indentifikohet dhe t'a kontrolloj profilin e tij. Nëse përdoruesi është kyçur me sukses në sistem me rolin e “Pacientit” ai do të ketë qasje në këtë modul, në të cilin kanë të integruar ndërfaqen e plotë e të personalizuar ku mund të menaxhojnë të gjithë sistemin e brendshëm, këto funksione janë të arritshëm vetëm nga paneli i pacientit.

5.1.3.1 Dashboard

Nëse përdoruesi është kyçur me sukses në sistem me rolin Pacient atëherë ai do të ketë qasje në modulën Dashboard i cili për këtë rol ofron të dhënat e përgjithshme të pacientit aktual në

një formë, e cila përmban informata siç është emri, mbiemri, email adresa, foto, moshë, gjinia, adresa, numri i kontaktit etj. Këto të dhëna mund të ndryshohen nga vetë pacienti.

Ky modul duket si në figurën më poshtë:



The image shows a web application interface for a patient's profile. On the left is a sidebar with the 'Medix' logo and navigation links for 'Dashboard' and 'Terminet'. The main content area is titled 'Profili' and features a profile picture of a man with a 'Zgjedh foton' button. Below the photo are input fields for 'Emri' (Name) with the value 'Fitim', 'Mbiemri' (Surname) with 'Mehmeti', 'Email adresa' (Email) with 'mehmetifitim92@gmail.com', 'Gjinia' (Gender) with a dropdown menu set to 'Mashkull', 'Adresa' (Address) with 'Shqipol, Mitrovica', 'Telefoni' (Phone) with '0491322654', and 'Moshë' (Age) with '20'. A blue 'RUAJ' (Save) button is at the bottom.

Figura 18. Dashboard për pacientin

Kodi i përdorur për dashboard:

```
public function patient_dashboard(int $patient_id): Patient
{
    // if no patient _id is assigned. Return an empty object!
    if(!$patient_id)
    {
        return new Patient;
    }
    $this->load-> model(' Patient ');
    $data['user'] = $this->Patients->get_patient ($patient_id);
    $this->load-> view('patient_dashboard', $data);
}
```

5.1.3.2 Terminet e pacientit

Nga menyja e majtë, duke klikuar në opsionin “Terminet” pacienti mund të shohë të gjithë listën e termineve të cilat i ka në dispozicion. Lista tregon datën, kohën, të dhënat e doktorit, statusin e terminit i cili nëse nuk është i përfunduar dhe nëse doktori është në linjë ofron mundësinë për fillim të video thirrjes duke klikuar butonin “starto thirrjen”.

Ky modul duket si në figurën më poshtë:

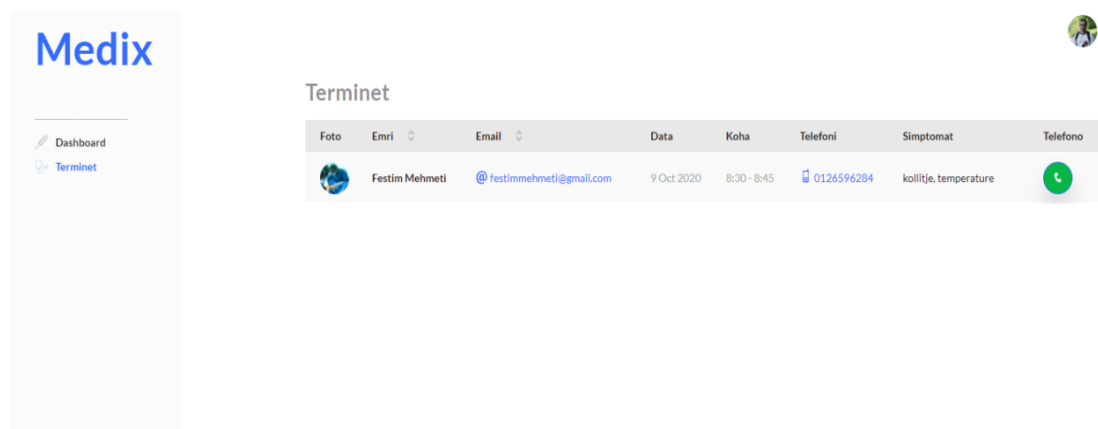


Figura 19. Terminet për pacientin

Kodi i përdorur për terminet:

```
public function appointments()
{
    $this->load->model('Appointments');
    $active_appointments = TRUE;
    $data[appointments]=$this->Appointments->
        get_appointments($active_appointments);
    $this->load->view('appointments', $data);
}
}
```

5.1.4 Moduli i doktorit

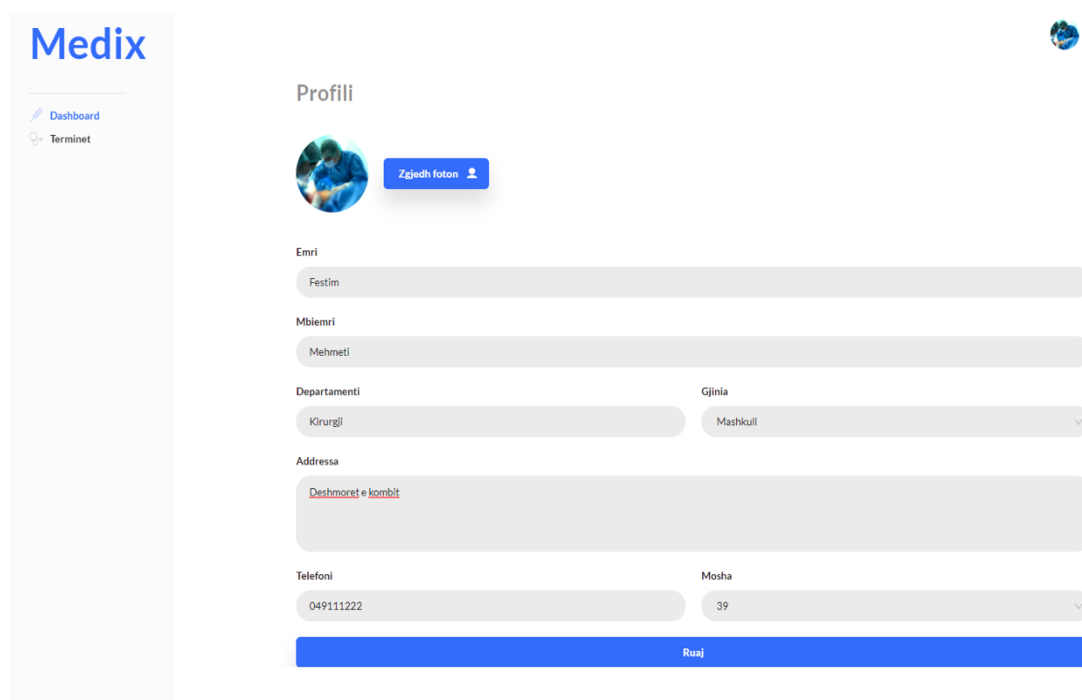
Nëse përdoruesi është kyçur me sukses në sistem me rolin e “Doktorit” ai do të ketë qasje në këtë modul, në të cilin kanë të integruar ndërfaqen e plotë të personalizuar ku mund të

menaxhojnë të gjithë sistemin e brendshëm, këto funksione janë të arritshme vetëm nga paneli i doktorit.

5.1.4.1 Dashboard

Nëse përdoruesi është kyçur me sukses në sistem me rolin “Doktor” atëherë ai do të ketë qasje në modulën Dashboard i cili për këtë rol ofron të dhënat përgjithshme të doktorit aktual në një formë e cila përmban informata siç është emri, mbiemri, foto e doktorit, numri kontaktues, mosha, gjinia, adresa, departamenti të cilit i takon etj. Këto të dhëna edhe mund të ndryshohen dhe të ruhen nga vetë doktori.

Ky modul duket si në figurën më poshtë:



The screenshot shows the Medix dashboard for a doctor. On the left is a sidebar with the Medix logo and navigation links for 'Dashboard' and 'Terminet'. The main content area is titled 'Profili' and contains a profile card with a photo placeholder and a 'Zgjedh foton' button. Below the photo are input fields for 'Emri' (Festim), 'Mbiemri' (Mehmeti), 'Departamenti' (Kirurgji), 'Gjinia' (Mashkull), 'Addressa' (Deshmoret e kombit), 'Telefoni' (049111222), and 'Mosha' (39). A blue 'Ruaj' button is at the bottom.

Figura 20. Dashboard për doktorin

Kodi i përdorur për dashboard:

```
public function doctor_dashboard(int $doctor_id): Doctor  
{
```

```

// if no doctor_id is assigned. Return an empty object!
if(!$doctor_id)
{
    return new Doctor;
}
$this->load-> model('Doctor ');
$data['user'] = $this->Doctors->get_doctor($doctor_id);
$this->load-> view('doctor_dashboard', $data);
}

```

5.1.4.2 Terminet e doktorit

Nga menyuja e majtë, duke klikuar në opsionin “Terminet” doktori mund të shohë të gjithë listën e termineve të cilat i ka në dispozicion.

Lista tregon datën, kohën, të dhënat e pacientit, statusin e terminit i cili nëse nuk është i përfunduar, ofron mundësinë për fillim të video thirrjes duke klikuar butonin “starto thirrjen”.

Ky modul duket si në figurën më poshtë:

Foto	Emri	Email	Data	Koha	Telefoni	Simptomat	Telefoni
	Afërdita	@aferdita@gmail.com	9 Oct 2020	8:30 - 8:45	0126596284	kolitje, temperature	
	Elita	@elita@gmail.com	9 Oct 2020	11:30 - 11:40	0126595200	hipotermi	

Figura 21. Terminet për doktorin

Kodi i përdorur për dashboard:

```
public function appointments()
```

```

{
  $this->load ->model('Appointments');
  $active_appointments = TRUE;
  $data[appointments]=$this-> Appointments->
      get_appointments($active_appointments);
  $this->load-> view('appointments', $data);
}
}

```

5.1.4.3 Video thirrja

Video thirrja është një opsion, që mundëson video komunikimin ndërmjet doktorit dhe pacientit.

Ky opsion bëhet i mundur pasi të jetë klikuar lista e termineve e cila e shfaq listën e pacientëve që kanë kërkuar termin me doktorin specifik, doktori ka mundësinë të klikoj butonin që i mundëson të realizojë thirrjen dhe të komunikoj me pacientin (shih figurën nr. 21). Në rastin kur doktori kërkon komunikimin me pacientin, aplikacioni duhet të kaloj nëpër disa hapa:

1. Doktori krijon një instance të klasës `RTCPeerConnection`.
2. Doktori krijon një ofertë për lidhje duke përdorur metodën `createOffer()` të klasës `RTCPeerConnection`.
3. Doktori përdor metodën `setLocalDescription()` me ofertën e krijuar me herët.
4. Doktori konverton ofertën në string dhe duke përdorur një mekanizëm për sinjalizim (në rastin tonë kemi përdorur `websocket`) e dërgon ofertën tek pacienti.

Kodi i përdorur për 4 hapat e parë:

```

const pc = new RTCPeerConnection(configuration);
pc.onnegotiationneeded = async () => {
  try {
    await pc.setLocalDescription(await pc.createOffer());
    signaling.send({desc: pc.localDescription});
  } catch (err) {
    console.error(err);
  }
}

```

```
}  
};
```

5. Pacienti përdor metodën `setRemoteDescription()` ku si parametër dërgon ofertën e pranuar nga Doktori, në mënyrë që objekti i klases `RTCPeerConnection` i krijuar në anën e tij të jetë në dijeni dhe në koordinim me konfigurimet e përdorura nga doktori.
6. Pacienti përdor `createAnswer()` e cila si parametër pranon ofertën e dërguar nga doktori, dhe krijon një sesion lokal i cili është kompatibil me atë të doktorit.
7. Pacienti vendos përgjigjen e krijuar me herët si përshkrim lokal duke përdorur metodën `setLocalDescription()`.
8. Pacienti pastaj konverton përgjigjen e tij në string dhe duke përdorur mekanizmin për sinjalizim e dërgon këtë përgjigje tek doktori.
9. Doktori vendos përgjigjen e pacientit si përshkrim për sesionin duke përdorur metodën `setRemoteDescription()`.

Kodi i përdorur për pranimin e ofertës, si dhe për pergaditjen e përgjigjes dhe dërgimin e saj tek bashkëbiseduesi:

```
signaling.onmessage = async ({desc, candidate}) => {  
  try {  
    if (desc) {  
      if (desc.type === 'offer') {  
        await pc.setRemoteDescription(desc);  
        const stream =  
          await navigator.mediaDevices.getUserMedia(constraints);  
        stream.getTracks().forEach((track) =>  
          pc.addTrack(track, stream));  
        await pc.setLocalDescription(await pc.createAnswer());  
        signaling.send({desc: pc.localDescription});  
      } else if (desc.type === 'answer') {  
        await pc.setRemoteDescription(desc);  
      }  
    } else if (candidate) {
```

```
    await pc.addIceCandidate(candidate);  
  }  
  });
```

Në të dy anët, në anën e doktorit edhe në atë të pacientit funksioni fillestar i cili thirret është funksioni `getUserMedia` i klasës `MediaDevices`, i cili mundëson qasje në audion dhe videon e paisjes së shfrytëzuesit. Sapo të thirret ky funksion shfrytëzuesit i'u shfaqet një dritare që kërkon të drejta për qasje në mikrofonin dhe kamerën e përdoruesit. Në momentin që lejohet qasja, metoda kthen një objekt të klasës `MediaStream` i cili mund të shfaqet në një tag të HTML. Ky proces i qasjes dhe i shfaqjes së zërit dhe videos është paraqitur në kodin më poshtë:

```
var getMedia = navigator.mediaDevices.getUserMedia({ audio: true, video: true });  
getMedia.then(function(mediaStream) {  
  var video = document.querySelector('video');  
  video.src = window.URL.createObjectURL(mediaStream);  
});  
p.catch(function(err) {  
  alert('Permission not granted');  
});
```

5.2 Teknologjitë e përdorura

Në këtë kapitull do t'i përshkruajmë dhe analizojmë teknologjitë që kemi përdorur për zhvillim të aplikacionit. Poashtu do të përshkruajmë edhe gjuhët programuese që janë përdorur për zhvillim.

5.2.1 Teknologjia webRTC

Web Real-Time Communication (WebRTC) filloi në vitin 2011 si projekt/përpjekje për mundësimin e komunikimit në kohë reale mes dy shfletuesve pa pasur nevojë për instalim të shtojcave në shfletues. WebRTC është zhvilluar dhe standardizuar nepermjet W3C dhe IETF së bashku me liderët e industrisë siç janë Google dhe Apple. Kjo teknologji është me “kod të hapur” (open source) dhe mund të përdoret nga secili zhvillues që dëshiron të zhvillojë aplikacion për komunikim në kohë reale. Në mënyrë që të realizohet një aplikacion me teknologjinë webRTC së pari duhet të realizohet lidhja mes shfletuesve që deshirojnë të komunikojnë, kjo lidhje fillestare realizohet nëpërmjet sinjalizimit. [8]

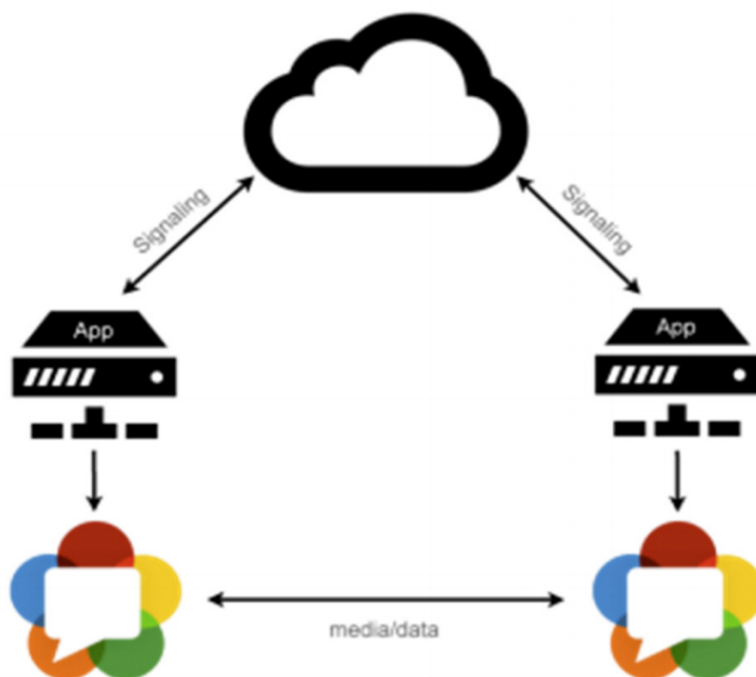


Figura 22. Trekendeshi webRTC

5.2.1.1 Sinjalizimi

WebRTC ofron zgjidhje për komunikim dhe transferim të të dhënave, por jo edhe për sinjalizim. Sinjalizimi është procesi i koordinimit të komunikimit mes klientëve që nuk gjenden në të njejtën rrjetë dhe duhet që t'i shkëmbejnë të dhënat bazike të cilat duhen për komunikim “peer to peer”. Në mënyrë që aplikacioni i zhvilluar me teknologjinë webRTC të

realizoj komunikimin, klientët duhet t'i shkëmbejnë disa informata nëpërmjet sinjalizimit, disa prej këtyre të dhënave janë:

- Te dhënat e rrjetës në të cilën gjenden, siç është IP adresa, porti etj.
- Te dhëna për video dhe zë, ku përfshihen: kodekët për video, bandwidth, të dhënat për paisjet që do të përdoren për video dhe zë (kamera dhe mikrofoni),
- Errorët që mund të ndodhin gjatë fillimit dhe procesit të komunikimit.

Sinjalizimi mund të implementohet në mënyra të ndryshme, pra mbetet në dorën e zhvilluesit që të bëjë zgjidhje për të. Një zgjidhje efiçente që mund të përdoret për sinjalizim është përdorimi i Websockets. Websockets krijojnë komunikim të dyanshëm mes klientit dhe serverit që do të thotë që ndërverprimi mes serverit dhe shfletuesit është “event-based”, pra shfletuesi nuk duhet të pret për përgjigje nga serveri çdo herë që duhet të shkëmbehen të dhëna, gjë që bën shumë të përshtatshme përdorimin e kësaj teknologjie për sinjalizim. [9]

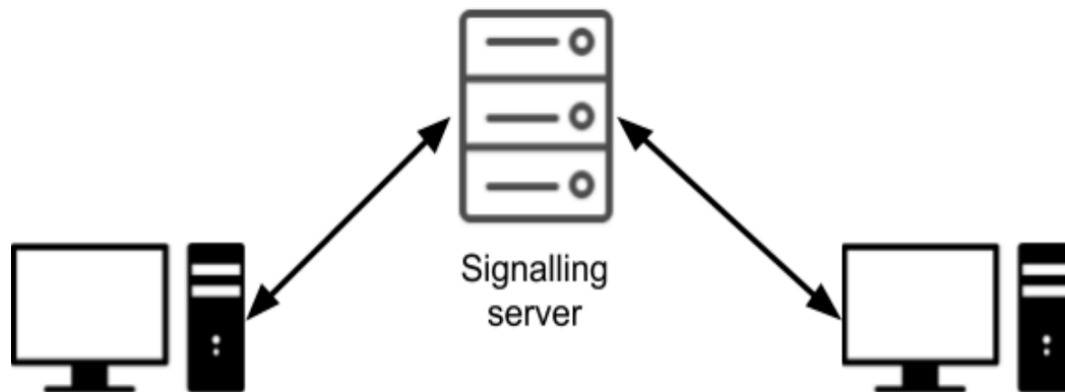


Figura 23. Shkëmbimi i mesazheve mes klientëve duke përdorur serverin për sinjalizim

5.2.1.2 ICE

Interactive Connectivity Establishment (ICE) provon të gjejë rrugën më të shkurtër për t'a realizuar lidhjen mes dy klientëve, së pari tenton t'a realizoj lidhjen direkt duke përdorur IP adresen private që është shkëmbyer nëpërmjet serverit për sinjalizim. Kjo lidhje realizohet vetëm nëse dy klientët janë brenda një rrjete të përbashkët, në rast që ky skenar nuk vlenë atëherë duhet të përdoret STUN serveri për t'a realizuar këtë lidhje duke përdorur IP adresen publike. [8]

5.2.1.3 Serveri STUN

Në rastet kur ICE dështon t'a realizoj lidhjen për komunikim duke përdorur IP adresën private, ose klienti nuk mund t'a identifikojë IP adresën publike të tij, nëse gjendet mbrapa një firewall ose NAT, atëherë webRTC rekomandon përdorimin e serverit STUN. Ky server mundëson nxjerrjen e informatave të nevojshme siç janë IP adresa publike edhe të dhëna se mbrapa çfarë lloji të NAT gjendet klienti. IP adresa publike nevojitet të shkëmbehet mes klientëve në mënyrë që ata të dijnë se ku duhen të dërgohen të dhënat për komunikim.

Pra roli i vetëm i serverit STUN është që të siguroj IP adresën publike të klientit. Në momentin kur dërgohet kjo IP adresë me sukses nëpërmjet sinjalizimit atëherë të dy klientët mund të komunikojnë me njëri-tjetrin. [8]

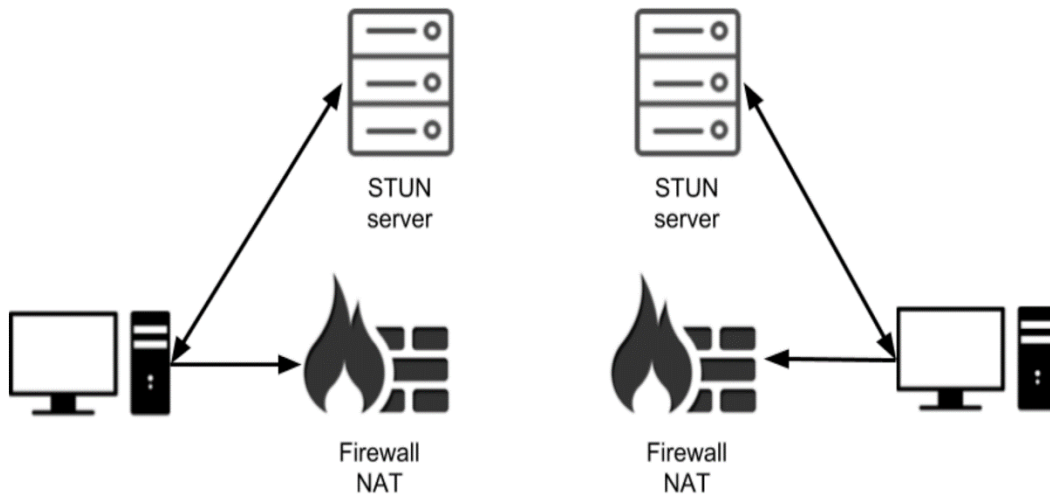


Figura 24. Klientet e pranojnë ip adresen publike nepermjet serverit STUN

5.2.1.4 Serveri TURN

Kur edhe STUN serveri dështon të realizoj komunikimin, në rastet kur dy adresat e klientëve janë mbrapa një firewall-i ose NAT, e që rezultojnë në pamundësi për të pranuar lidhjen nga ana e klienteve, atëherë duhet të përdoret TURN serveri i cili shërben si server bartës pasi që gjendet në internet dhe është publik për qasje nga të dy klientët. Të dhënat prej adresës së njerit klient dërgohen së pari në TRUN server, i cili pastaj i dërgon ato tek adresa e klientit tjetër. [8]

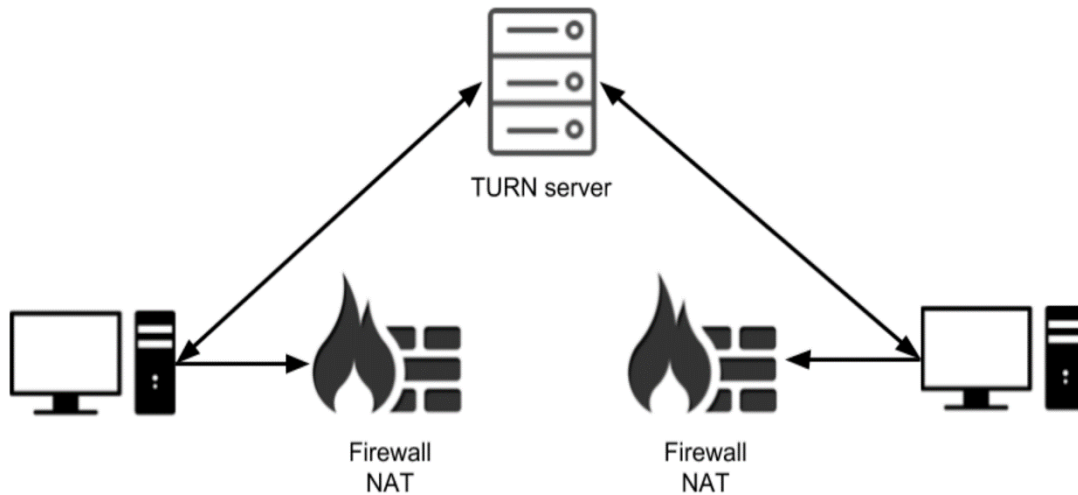


Figura 25. Te dhenat e dy klienteve barten nepermjet serverit TURN

5.2.2 WebSocket

WebSocket është protokoll kompjuterik i komunikimit, ky protokoll siguron lidhje të vazhdueshme mes klientit dhe serverit, ky protokoll ofron komunikim të dyfishtë në një lidhje të vetme që operon përmes HTTP-s. Pra veçoria bazë e websocket është lehtësimi i kalimit të mesazheve midis klientit dhe serverit pas pasur nevojë të dërgohen kërkesa dhe përgjigje çdo herë.

Ndryshe nga HTTP e cila fillon me `http://` ose `https://`, websocket fillon me `ws://` ose `wss://`. Ky protokoll siguron që lidhja midis klientit dhe serverit do të mbahet gjallë derisa të ndërpritet nga njëra palë (klienti ose serveri), pas mbylljes së lidhjes nga cilido lidhja ndërpritet në të dyja skajet. [10]

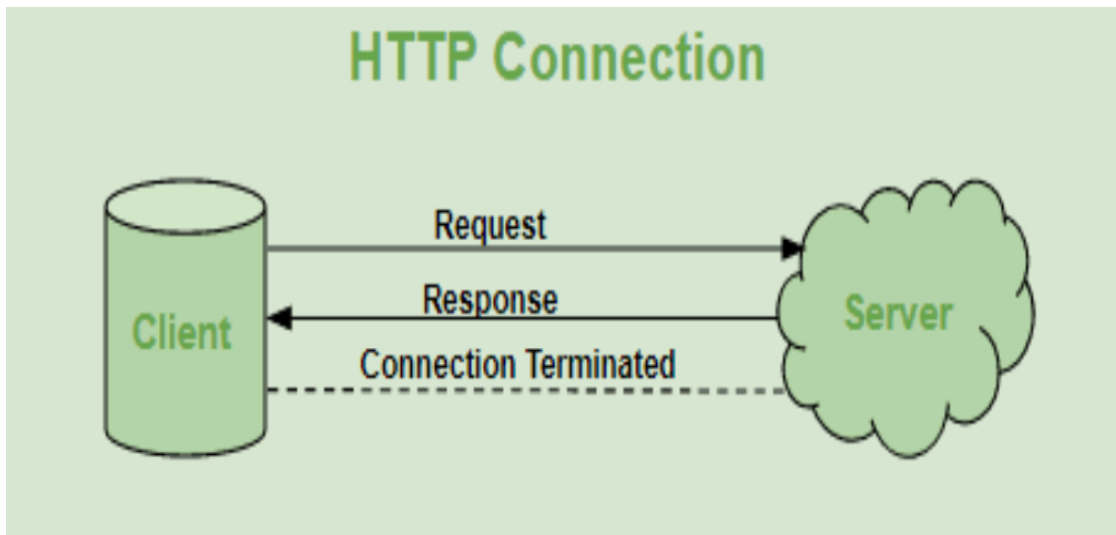


Figura 26. Komunikimi në websocket

Websocket gjen zbatim në raste dhe fusha të ndryshme, disa prej tyre janë:

Web aplikacionet në kohë reale - Aplikacionet në kohë reale përdorin websocket për të shfaqur të dhënat më të fundit tek klientit, këto të dhëna vazhdimisht dërgohen nga serveri, kjo gjë ndihmon në përmirësimin e performancës së aplikacionit, dhe është shumë e dobishme në aplikacionet për tregëti (trading) ku çmimet ndryshojnë shumë shpesh. [11]

Aplikacionet për komunikim – aplikacionet për komunikim siç është rasti i sistemit që ne kemi zhvilluar, e përdorin websocket për të realizuar vetëm një lidhje për shkëmbim të të dhënave, pra kjo lidhje përdoret gjatë gjithë komunikimit për dërgim dhe pranim të mesazheve, kjo do të thotë se websocket është adekuat për realizimin e mekanizmit për sinjalizim. [11]

5.3 Gjuhët Programuese

Në pjesën në vijim do t'i përshkruajmë gjuhët programuese që ne i kemi përdorur për zhvillimin e aplikacionit për komunikim me pacientë.

5.3.1 HTML

HTML (HyperText Markup Language) – është gjuhë e strukturuar dhe është njëra nga gjuhët kryesore që përdoret për të krijuar ueb faqe të ndryshme në internet varësisht nga preferencat e përdoresit. HTML përdor etiketat, të cilat përdoren për të përshkruar dhe strukturuar faqet në internet. Etiketat e HTML mund të identifikohen lehtë në një dokument HTML pasi ato fillojnë “<” dhe mbarojnë me “>”, brenda etiketave të html-s mund të vendosen edhe atributet për etiketën, atributet janë informata shtesë për etiketën/elementin ku janë vendosur. HTML na ofron gjithashtu mundësinë e krijimit të dokumenteve multimediale, me dokument multimedial nënkuptojmë dokumentin i cili përmban: imazhe, tinguj, video etj. [12]

5.3.2 CSS

CSS (Sheets Cascading Style) - është një gjuhë që përshkruan pamjen dhe formatimin e dokumentit të shkruar në gjuhën HTML, pra kjo gjuhë përdoret gjerësisht për të stiluar faqet në internet.

CSS mundëson ndarjen e përmbajtjes së dokumentit nga prezantimi i dokumenteve, siç janë paraqitja, ngjyrat dhe shkronjat. Kjo mundëson që shumë faqe të internetit të ndajnë të njëjtin formatim dhe të zvogëlojnë kompleksitetin dhe përsëritjen e përmbajtjes strukturore. CSS gjithashtu mund të lejojë që e njëjta faqe në HTML të paraqitet në stile të ndryshme.

Sot CSS përdoret dhe mbështetet gjerësisht nga të gjithë shfletuesit. [13]

5.3.3 PHP

PHP (Hypertext Preprocessor) është një gjuhë programuese e lehtë dhe shumë e përshtatshme për t'u përdorur e cila funksion primar e ka zhvillimin e ueb faqeve. Php për nga sintaksa ka ngjashmëri me shumë gjuhë tjera si C, Perl, Java etj.

Kjo gjuhë mund të bashkëveprojë me lloje të ndryshme bazash të të dhëva, si p.sh.: MySQL, MSSQL etj.

Skriptat e PHP fillon me “<?php” dhe perfundon me “?””. Me anë të kësaj gjuhe programerit i mundësohet dhe e ka më të lehtë të ndërmjetsojë mes HTML apo anës së klientit dhe anës së serverit ku ruhen të dhënat. Pasi që PHP ekzekutohet në server, atëherë klienti nuk mund të ketë qasje në kod. [14]

Programet e shkruara në PHP nuk kompajlohen por interpretohen, gjë që e bënë këtë gjuhë më të ngadaltë.

Disa prej vetive kryesore të PHP-së janë:

- PHP është open source, dmth falas për përdorim,
- PHP është një gjuhë e skriptuese,
- PHP-së ekzekutohet në anën e serverit,
- PHP mbështet shumë lloje të bazës së të dhënave (MySQL, MSSQL, Oracle, Sybase etj.),
- PHP punon në platforma të ndryshme (Windows, Linux, Unix),
- PHP është kompatible me pothuajse të gjithë ueb serverët, të cilët përdoren sot (Apache, IIS etj.)
- Një PHP file mund të përmbajë tekst të thjeshtë, HTML tags dhe skripta,
- File-at e PHP mund të kenë njërin nga këto prapashtesa: php, php3, phtml etj.,
- PHP përkrah klasat edhe objektet,
- PHP përkrah sesionet dhe cookiet,
- Variablat në PHP kanë tipe dinamike (nuk ka nevojë t’u deklarohet tipi). [13]

5.3.4 Javascript

Javascript është një gjuhë programuese e zhvilluar nga Netscape në vitin 1995. Kjo gjuhë është gjuha e vetme skriptuese në anën e klientit, e cila përdoret në cdo web faqe në internet. Meqenëse JavaScript ekzekutohet në shfletues, nuk ka nevojë për softuerë shtesë për t’a përdorur, duhet vetëm një tekst editor (si TextWrangler apo TextEdit ose NotePad) si dhe një shfletues për përdorimin e kësaj gjuhe.

Javascript nuk ka lidhje me gjuhën programuese Java edhe pse të dyja kanë ngajshmëri sipërfaqësore.

Kjo gjuhë mund të ndërveprojë me HTML duke ju lejuar ueb dizajnerëve të ndërtojnë ueb faqe me një përmbajtje dinamike. Kodi JavaScript vendoset brenda tag-eve <SCRIPT>. Këto tag-e mund të vendosen në çdo pjesë të dokumentit. Browserat të cilët nuk suportojnë kod JavaScript thjeshtë do t'a injorojnë përmbajtjen e këtyre tag-eve. Zakonisht tag-et <SCRIPT> vendosen në kokën e faqes në menyrë që kodi HTML të jetë sa më i lexueshëm.

JavaScript mund të përdoret për të ndërtuar një uebfaqe të plotë nëse përdoret me Node.js (librari që përdoret në anën e serverit) dhe MongoDB (që është një bazë të dhënash). [14]

5.3.5 Baza e të dhënave

Baza e të dhënave është një vend ku ruhen të dhënat në mënyrë të organizuar. Aplikacionet ueb ndërveprojnë me të dhënat e vendosura në një databazë të caktuar në tabela të ndryshme vetëm pasi që të jetë bërë lidhja mes tyre. Në sistemin që ne kemi zhvilluar për bazën e të dhënave është përdorur MySQL.

Ekzistojnë dy forma për t'u lidhur me databazë:

Lidhja automatike - kjo mënyrë mundëson qasje në çdo ngarkim të klasës. Për t'a aktivizuar këtë veçori, duhet të shtohet fjala “database” tek array i librarive, në file-n: application/config/autoload.php. [15]

Lidhja manuale - nëse vetëm disa faqe kërkojnë lidhje në bazë të të dhënave atëherë përdoret kjo mënyrë e cila mundëson qasje në bazë të të dhënave, duke përdorur rreshtin e kodit aty ku duhet qasja, ose në konstruktor për të mundësuar çsajë globale në të gjithë klasën. Për t'u lidhur manualisht me një bazë të dhënash, duhet të dërgohet ky varg vlerash: [15]

```
$config['hostname'] = 'localhost';  
$config['username'] = 'myusername';  
$config['password'] = 'mypassword';  
$config['database'] = 'mydatabase';  
$config['dbdriver'] = 'mysqli';  
$config['dbprefix'] = '';  
$config['pconnect'] = FALSE;  
$config['db_debug'] = TRUE;  
$config['cache_on'] = FALSE;  
$config['cachedir'] = '';
```

```
$config['char_set'] = 'utf8';  
$config['dbcollat'] = 'utf8_general_ci';  
$this->load->database($config);
```

5.4 Kërkesat e sistemit

Kjo pjesë ka të bëjë me përshkrimin e kërkesave mbi të cilat do të zhvillohet sistemi. Këto kërkesa janë ndarë në dy grupe, në kërkesa funksionale dhe kërkesa jofunksionale.

5.4.1 Kërkesat funksionale

- Sistemi duhet të ofroj mundësinë e regjistrimit të përdoruesve.
- Sistemi duhet të ofroj mundësinë për qasje në sistem.
- Sistemi duhet të ofroj mundësinë e regjistrimit të reparteve.
- Sistemi duhet të ofroj mundësinë e regjistrimit të termineve.
- Sistemi duhet të ofroj mundësinë e shfaqjes së termineve për doktorët dhe pacientët.
- Sistemi duhet të ofroj mundësinë për realizimin e video thirrjeve.
- Sistemi duhet të ofroj mundësinë për përditësimin e të dhënave të përdoruesit.
- Sistemi duhet të ofroj mundësinë e deaktivizimit të përdoruesve të sistemit.

5.4.2 Kërkesat jo-funksionale

- Sistemi duhet të jetë i bazuar në teknologjinë ueb.
- Sistemi duhet të ofroj përkrahje 24/7.
- Sistemi duhet të jetë i qasshëm nga çdo pajisje si kompjuter, telefon të mençur, tablet etj.
- Sistemi duhet të zhvillohet me CodeIgniter framework 4.0.
- Sistemi duhet të përdor teknologjinë webRTC për komunikim në kohë reale,
- Sistemi duhet të ofroj siguri të lartë për mbrojtjen e të dhënave.
- Sistemi duhet të jetë responsiv për çdo shfletues.
- Sistemi duhet të ofroj role të ndryshme për përdoruesit.
- Sistemi duhet të përkrahë shumëgjuhësinë.

6 DISKUTIME DHE PËRFUNDIME

Në këtë punim diplome qellimi ynë ka qenë hulumtimi se si punon një sistem i telemjekësisë, gjithashtu ne kemi propozuar një sistem i cili mund të ketë një rol të rëndësishëm në aspektin e kujdesin shëndetsor.

Meqenëse tipari kryesor i sistemit tonë është thirrja me video, ne u përpoqëm të japim trajtim të menjëhershëm që të zvogëlohet mbingarkesa e spitaleve në vendin tonë.

Përmes komunikimit video-thirrje mjeku, komunikon me pacientin, duke u njoftuar me të gjitha problemet me të cilat pacienti përballet, mund të përshkruaj ilaçet të nevojshme dhe gjithashtu të monitorojë semundjen e pacientit dhe nëse është e nevojshme, mjeku të rekomandojë të vizitojë fizikisht.

Gjithashtu kemi krahasuar tri sistemet (doxy.me, eVisit dhe teladoc) te cilat konsiderohen që përdoren më së shumti në botë , ku pas krahasimeve të funksioneve bazë del që Teladoc është e vetmja që ofron aplikacionet mobile ne ios edhe android. Gjithashtu Teladoc konsiderohet si sistemi me çmimin më të lartë në krahasim me sistemet e tjera.

Për zhvillimin e këtij sistemi ne kemi përdorur gjuhët programuese si: PHP (CodeIgniter), HTML5, CSS3, Javascript, jQuery, webRTC, websocket, ndërsa baza e të dhënave e aplikacionit është e ndërtuar në MySQL, sistemi sinjalizues është zhvilluar duke përdorur websocket dhe gjuhën javascript.

Sistemi për komunikim me pacientë ka tri role kryesor: Stafin mjekësorë, Doktorët dhe Pacientët, per secilin rol është zhvilluar një nënsistem në vete.

Në këtë sistem përpos moduleve bazë të zhvilluara ka mundësi që në të ardhmen të shtohen edhe role dhe module të reja, siç është psh.: komunikimi i aplikacionit me sensor të ndryshëm mjekësor dhe bartja e të dhënave të sensorit në server për t'a ditur më saktë gjendjen aktuale të pacientit.

7 REFERENCAT

- [1] «World Health Organization,» [Në linjë]. Available: <https://www.who.int/gho/goe/telehealth/en/>. [Qasja 10 8 2020].
- [2] «Agjencia e Statistikave të Kosovës,» [Në linjë]. Available: <https://ask.rks-gov.net/media/4892/vler%C3%ABsimi-i-popullsis%C3%AB-2018.pdf>. [Qasja 22 8 2020].
- [3] «Shërbimi spitalor dhe klinik universitar i kosovës,» [Në linjë]. Available: <https://shskuk.rks-gov.net/Navbar/IndexStatic/1047>. [Qasja 11 9 2020].
- [4] «Shërbimi spitalor dhe klinik universitar i kosovës,» [Në linjë]. Available: <https://shskuk.rks-gov.net/Navbar/IndexStatic/1047>. [Qasja 12 8 2020].
- [5] «Doxy,» [Në linjë]. Available: <https://doxy.me/en/about/>. [Qasja 20 8 2020].
- [6] «eVisit,» [Në linjë]. Available: <https://evisit.com/platform/overview/>. [Qasja 20 8 2020].
- [7] «Teladoc,» [Në linjë]. Available: <https://www.teladoc.com/>. [Qasja 20 8 2020].
- [8] S. Loreto dhe S. P. Romano, Real-Time Communication with WebRTC: Peer-to-Peer in the Browser, 2014.
- [9] «MDN Web Docs,» [Në linjë]. Available: https://developer.mozilla.org/en-US/docs/Web/API/WebRTC_API/Signaling_and_video_calling. [Qasja 25 8 2020].
- [10] A. Lombardi , WebSocket: Lightweight Client-Server Communications, 2015.
- [11] «Geeks for Geeks,» [Në linjë]. Available: <https://www.geeksforgeeks.org/what-is-web-socket-and-how-it-is-different-from-the-http/>. [Qasja 24 8 2020].
- [12] «W3C,» [Në linjë]. Available: <https://www.w3.org/MarkUp/Guide/>. [Qasja 21 8 2020].
- [13] «Mozilla Developers,» [Në linjë]. Available: <https://developer.mozilla.org/en-US/docs/Web/CSS>. [Qasja 21 8 2020].
- [14] R. Nixon, Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5, 2018.
- [15] «Codeigniter Web Framework,» [Në linjë]. Available: <https://codeigniter.com/userguide3/database/connecting.html>. [Qasja 23 8 2020].