

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Tristan Krass 179302IADB

**RAKENDUS KÕNEHÄIRETE
DIAGNOSTIKA JA KAUGTERAAPIA
LÄBIVIIMISE TOETAMISEKS**

Bakalaureusetöö

Juhendaja: Märt Kalmo
MSc

Tallinn 2021

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Tristan Krass

15.05.2021

Annotatsioon

Antud bakalaureusetöö raames antakse ülevaade olemasolevast rakendusest ning selle puudustest ja kirjeldatakse loodavale rakendusele esitatavaid ärinõudeid. Töö eesmärgiks on välja selgitada, milline on parim tehniline lahendus ning luua uus rakendus.

Antud lõputöö käsitleb ärinõuete tõlkimist tehnilisteks nõueteks. Seejärel analüüsitakse neid, et valida kõige sobivam lahendus antud kontekstis. Parim lahendus valitakse analüüsisides erinevaid võimalusi ning nende tugevusi ja nõrkusi. Lisaks võetakse arvesse kliiniku võimalusi ja piiranguid.

Tehniliste nõuete analüüsile järgneb praktiline teostus. Viiendas peatükis kirjeldatakse, kuidas teostati kõige olulisemad võimalused rakenduses.

Viimaks võetakse kokku töö tulemused. Autor annab hinnangu, kas töö alguses seatud eesmärgid saavutati. Lisaks saadakse tagasisidet kliiniku töötajatelt, kes hindavad loodud rakendust.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 30 leheküljel, seitse peatükki, 11 joonist, üks tabel.

Abstract

Application for Providing Telehealth and Support to the Detection of Speech, Language and Communication Disorders

The final thesis gives overview of the current application its features and limitations. Business use cases are then described for the new application. The goal of this thesis is to find the best technical solution and then implement it.

This thesis outlines the process of translating the business use cases to technical requirements. Technical requirements are then analysed to choose the most appropriate technology, considering the clinic's restrictions. The most appropriate technical solution is chosen by analysing different possibilities.

After the technical requirements are set, the actual application will be built. Chapter 5 will show how most important use cases were built and implemented.

Finally, the thesis will conclude the results. This will be done by getting feedback from the clinic personnel. Lastly, the author reflects on how well the solution was built.

The thesis is in Estonian and contains 30 pages of text, seven chapters, 11 figures, one table.

Lühendite ja mõistete sõnastik

API	<i>Application programming interface</i> , rakendusliides
CI	<i>Continuous integration</i> , pidev integratsioon
DAO	<i>Data access object</i> , andmete juurdepääsu disaini muster
Docker compose	Tööriist konteinerite defineerimiseks ning käivitamiseks
Docker container	Konteiner, mille sees jookseb rakendus, mis on ehitatud docker image põhjal
Docker image	Mall, mille järgi ehitatakse konteinerid
Git	Versioonihaldustarkvara
HTTP	<i>Hypertext Transfer Protocol</i> , veebiprotokoll andmete edastamiseks üle võrgu
Joomla	Sisuhaldussüsteem
JSON	<i>JavaScript Object Notation</i> , andmevahetusvorming
JWT	<i>JSON Web Token</i> , andmevahetusvorming lubade hoiustamiseks
Logopeediakliinik	Erinevaid tervishoiuteenuseid pakkuv ettevõtte Eestis
Object storage	<i>Computer data storage architecture</i> , andmete salvestamise arhitektuur
OpenAPI	Standard veebiteenuste kirjeldamiseks
OWASP	<i>Open Web Application Security Project</i> , veebirakenduste turvalisuse projekt
PWA	<i>Progressive Web App</i> , progressiivne veebirakendus
SDK	<i>Software development kit</i> , tarkvaraarenduse komplekt
SK ID Solutions AS	Ettevõtte, mis pakub autentimisteenust
SMS	<i>Short message service</i> , tekstisõnumite saatmise teenus telefonidel
SPA	<i>Single-page application</i> , üheleherakendus
SQL	<i>Structured Query Language</i> , struktuuripäringukeel
Tugispetsialist	Logopeed, eripedagoog, psühholoog
URL	<i>Uniform Resource Locator</i> , veebiaadress
WebRTC	<i>Web Real-Time Communications</i> , reaajas suhtlus
YAML	<i>YAML Ain't Markup Language</i> , andmete jadastuse standard

Sisukord

1 Sissejuhatus	10
2 Probleemi püstitus ja eesmärk	11
2.1 Aktuaalsus	11
2.2 Taust	11
2.3 Probleemi püstitus	12
2.4 Eesmärk	12
2.5 Metoodika	13
3 Lähtetingimused	14
3.1 Hetkeolukord	14
3.2 Ärinõuded	15
3.2.1 Konto loomine	15
3.2.2 Ülesannete üleslaadimine	15
3.2.3 Tagasiside ülesannetele	15
3.2.4 Spetsialisti vaade rakenduses.....	16
3.2.5 Harjutusvideote vaatamine	16
3.2.6 Videokõne võimalus	16
3.2.7 Mitme keele tugi.....	16
4 Analüüs.....	17
4.1 Rakenduse turvalisus	17
4.1.1 Turvameetmed	17
4.1.2 Autentimine ja autoriseerimine rakenduses.....	19
4.2 Failide salvestamine	21
4.3 Videomängija lahenduse analüüs	23
4.3.1 Olemasolevad lahendused turul.....	23
4.3.2 Videomängija serveripoolne lahendus.....	23
4.3.3 Teavituste saatmine	24
4.4 Videokõne võimalus	24
4.5 Tehnoloogiate valik	26
4.5.1 Serveripoolne rakendus	26

4.5.2 Esilehe rakendus	27
4.5.3 Andmebaasi haldussüsteemi valik	27
4.6 Arendusprotsess	28
4.6.1 Pidev integratsioon	28
5 Praktiline lahendus	30
5.1 Rakendusliidese kirjeldamine ja kasutamine	30
5.1.1 Probleem OpenAPI kasutamisega	30
5.2 Autentimine rakenduses	31
5.2.1 Autentimine toodangu- ja arenduskeskkonnas	31
5.3 Failide salvestamine	32
5.4 Video- ja helifailide mängija lahendus	33
5.5 Kaugteraapia pakkumise võimalus	34
5.6 Mitme keele tugi	35
5.7 Pidev integratsioon ja evitamine	36
6 Tulemused	37
6.1 Tagasiside	37
6.2 Tulevikuplaanid	37
7 Kokkuvõte	39
Kasutatud kirjandus	40
Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks	43
Lisa 2 Spetsialistide vaade failidest	44
Lisa 3 Videokõne vaade rakenduses	45
Lisa 4 Patsiendi tervikprofiili nägemise võimalus	46

Jooniste loetelu

Joonis 1. Päringute logimine.	19
Joonis 2. Autentimise voog.	20
Joonis 3. Videomängija andmemudel.	24
Joonis 4. Järgnevuskeem – kasutaja autentimine.	31
Joonis 5. Autentimine rakenduses sõltuvalt seadistusele.	32
Joonis 6. Failide valideerimine brauseris.	33
Joonis 7. Signeeritud veebiaadressi geneerimine tagarakenduses.	34
Joonis 8. Kaugteraapiaga liitumine rakenduses.	34
Joonis 9. Videokõne seadmistamise võimalus.	35
Joonis 10. I18n konfiguratsioon.	35
Joonis 11. Docker compose fail serveri ja kliendi käivitamiseks.	36

Tabelite loetelu

Tabel 1. Pilveteenuse pakkujate võrdlus.	21
---	----

1 Sissejuhatus

Järk-järgult on tehnoloogia muutunud aina olulisemaks inimeste igapäeva elus. Kindlasti on seda osakaalu muutnud COVID-19 pandeemia, mistõttu suur osa tegevustest on viidud üle digikanalitele. Teisalt on lihtsalt mugavam osa asju teha virtuaalselt.

Muutus pole puudutamata jätanud ka logopeede ja eripedagooge, kes täna erinevaid digilahendusi kasutavad, et teha oma tööd paremini ja efektiivsemalt. Logopeediakliinik on otsustanud olemasoleva rakenduse välja vahetada uue vastu, mis võimaldaks kogu raviprotsessi virtuaalselt läbi viia.

Käesoleva bakalaaurusetöö raames aitab autor Logopeediakliinikul välja selgitada uue ehitatava rakenduse tehnilised nõuded ning kirjutab valmis esialgsetele nõuetele vastava rakenduse. Lisaks soovib kliinik olla eesrindlik ja eeskuju enda valdkonnas ning arendada eriala laiapõhjalisust. Rakenduse väljaehitamine annab selleks lisavõimalusi.

Töö on jaotatud neljaks: lähtetingimused, analüüs, lahendus ja tulemused. Lähtetingimustes kirjeldatakse kliinikus olevaid protsesse ja kasutatavaid tööriistu ning tuuakse välja ärinõuded uuele rakendusele. Sellele järgneb tehniline analüüs, kus selgitatakse välja, mis on kõige sobivamad tehnoloogiad ja teenused antud olukorras. Seejärel kirjutatakse lahenduse teostamisest ning ette tulnud probleemidest. Viimaks sõnastatakse töö tulemused ja võetakse kokku tööle antud tagasiside.

2 Probleemi püstitus ja eesmärk

Käesolevas peatükis kirjeldatakse teema aktuaalsust, olemasolevat lahendust ja tausta. Tuuakse välja lahendatav probleem ning kirjeldatakse probleemi uurimiseks ja lahendamiseks valitud metoodikat. Viimaks tuuakse välja töö eesmärgid.

2.1 Aktuaalsus

Seoses globaalse pandeemiaga on muutunud digilahenduste olemasolu vajalikuks kõigile tervishoiuasutustele [1]. Eesti Haigekassa hakkas kaugteenust osa erialade puhul rahastama 2020. aasta sügisest ning tugispetsialistide, kellele antud lõputöö raames rakendus ehitatakse, puhul 2021. aasta jaanuarist [2]. Paralleelselt kuulutati välja projektikonkursid kaugteenuse turvalisemaks ja mugavamaks kasutamiseks. Eestis pole hetkel sobivat terviklahendust logopeedidele, mis soodustaks igapäeva tööd ning võimaldaks pakkuda kaugteraapiat.

2.2 Taust

Tervishoiusüsteemis rakendatakse järjest enam erinevaid digitehnoloogilisi võimalusi, sealhulgas esmaste konsultatsioonide läbiviimist ning diagnostikat videosilla teel. Järjest võimekam ja kvaliteetsem tehnoloogia pakub võimalust muuta tervishoiuteenused patsiendile kättesaadavamaks, vähendada transpordikuluseid ja vastuvõttude tühistamise hulka. Mõningatel juhtudel võib tehnoloogia kasutuselevõtt parandada ravi (nt teraapiatulemused kanduvad tavaellu paremini üle, sest teraapiat rakendatakse koduses keskkonnas), tõsta huvi ja motivatsiooni [3], [4]. Mujal maailmas on kaugteraapiat logopeedilise teenuse osutamiseks rakendatud aktiivselt juba üle paarikümne aasta, Eestis on seda siiani valdavalt kasutatud erapraksistes, alates 2021. aasta algusest rahastab teenust ka Eesti Haigekassa [2].

Logopeediakliinik, kes on teenuse parema kättesaadavuse ja kõrgema kvaliteedi tagamiseks kasutanud digitehnoloogilisi lahendusi alates 2017, soovib olemasolevat

kaugteenuse lahendust muuta mugavamaks, kasutajasõbralikumaks, hinnata turvariske ning vajadusel nendes korrekture teha.

2.3 Probleemi püstitus

Hetkel kasutab Logopeediakliinik Joomla baasil tehtud lahendust, mis esmalt oli mõeldud kodulehena ja tänaseks on ehitatud juurde võimalus kasutajatel videoid üles laadida ning logopeedidel videotele hinnanguid anda. Arvestades, et rakendus on eelkõige kliiniku koduleht, kus administraator saab juurde lisada erinevaid pistikprogramme, on praegune lahendus kõrge turvariskiga. Rakenduses puuduvad ka selgelt eraldatud kasutajate rollid.

Rakenduse arendusprotsess on varasemalt toimunud ilma versioonihalduseta ja rakenduse evitamine toimub manuaalselt. Selline lähenemine suurendab võimalust, et muutus võib rakenduse katki teha.

Lisaks on tekkinud vajadus uute võimaluste järele ning rakenduse praegune arhitektuur ei võimalda neid täita. Logopeedidel pole võimalik grupeerida, otsida, järjestada ega sildistada patsientide poolt üles laaditud video- ja helifaile. Puudub võimalus pakkuda kaugteraapiat, mida oleks vajadusel logopeedil võimalik salvestada. Viimaks puudub võimalus konsiiliumi läbiviimiseks, et keerukamate juhtumite korral diagnostikat ja raviplaani täpsustada.

2.4 Eesmärk

Käesolevas töös on kaks eesmärki:

1. Analüüsida olemasoleva rakenduse puuduseid ning leida tehniline lahendus, mis kõrvaldab puudused ja on jätkusuutlikult arendatud
2. Luua uus rakendus

Analüüsist lähtudes töötatakse välja uus lahendus, mis kõrvaldab olemasoleva rakenduse probleemid ja puudused. Võimalike lahenduste leidmiseks võetakse aluseks rakendusele esitatavad nõuded.

Lahenduse väljatöötamisel on oluline, et uus rakendus oleks jätkusuutlikult arendatud ning võimaldaks tulevikus mõnel muul ettevõttel vajadusel arendustegevust jätkata.

2.5 Metoodika

Töös sõnastatakse lahendatav probleem, lähtetingimused ja hetkeolukord. Rakenduse tehnilise analüüsi teostamisel lähtutakse äpile esitavatest nõuetest. Lisaks sellele tuleb arvestada Tervishoiuteenuste korraldamise ja Isikuandmete kaitse seadustega.

Sobiva lahenduse valimiseks tuuakse välja erinevad võimalused probleemi lahendamiseks ning analüüsitakse, milline neist on parim antud kontekstis. Seejärel arendatakse uus rakendus.

Rakendust ehitatakse järk-järgult osade kaupa. Iga faasi lõpuks on valmis arendatud midagi, mille saab koheselt kasutusele võtta. Tehtud rakendusele antakse logopeedide poolt tagasisidet. Selle põhjal antakse hinnang, kui hästi toimib väljatöötatud lahendus.

3 Lähtetingimused

Antud peatükis antakse esmalt ülevaade olemasolevast rakendusest. Seejärel tuuakse välja uuele rakendusele esitatud ärinõuded ning kirjeldatakse nende sisu.

3.1 Hetkeolukord

Hetkel on kõnealuses asutuses kasutusel minimaalne lahendus, mis võimaldab patsiendi ja spetsialisti vahel asünkroonset info edastamist video ja heli kujul.

Rakendus loob väärtust nii logopeedidele kui patsientidele, kuna tagab ravi järjepidevuse ja kvaliteedi. Digitaalse kaugteenuse kasutamisel on ilmnenud mitu puudust, mis uues rakenduses tuleb kindlasti lahendada.

Olemasolev rakendus on arendatud Joomla sisuhaldustarkvarale. Nimetatud tarkvara võimaldab pistikprogramme arendades laiendada tuumikvõimalusi. Praegune rakendus ongi lahendatud pistikprogrammina.

Kasutajaid hallatakse läbi Joomla süsteemi. Patsiendil pole võimalik endale ise kasutajat veebis luua. Kui logopeed peab vajalikuks, et patsient kas enne esmast kohtumist või teraapiate vahepeal lindistaks kodus teatud sooritusi või situatsioone ning laadiks salvestise portaali, peab ta administraatorile edastama konto avamise teatise patsiendi isikuandmetega. Administraator logib seejärel rakendusse sisse ja loob patsiendile kasutaja. Patsient saab kinnituse meilile ja seejärel on tal võimalik ID-kaardi või Mobiil-ID abil portaali sisse logida. Rakenduses on patsiendil võimalik video- ja helifaile üles laadida, muuta ja kustutada. Logopeedidel on veebilehele sisse logides võimalik kõikide patsientide videoid näha.

Tulenevalt sellest, et sisselogimine on ehitatud Joomla tuumikprogrammi, siis uuenduste käigus võib sisselogimise võimalus lakata töötamast. Koduleht on evitatud kliiniku enda serverisse. Rakendust uuendatakse ja muudetakse otse serveris, mis on vigade aldis. Hetkel ei kasutata rakenduse täiendavate võimaluste arendamiseks versioonihaldustarkvara, mistõttu uuenduste käigus tekkida võiva vea korral pole

võimalik algset seisut taastada. Viimaks puuduvad rakendusel ühik- ja integratsioonitesti.

3.2 Ärinõuded

Järgnevatel alapeatükkides antakse ülevaade uuele rakendusele esitatud ärinõuetest.

3.2.1 Konto loomine

Konto loomine peab olema võimalikult mugav ning turvaline. Sellest lähtuvalt rakendatakse kasutajate autentimiseks Mobiil-ID ja Smart-ID liidestust. Mainitud tehnoloogiad soodustavad tulevikus teenuse sidumist e-tervisega.

Suur osa kliiniku patsientidest on lapsed, kes ise kontot avada ja hallata ei saa. Lapsevanemal või hooldajal peab olema võimalus luua lapsele konto, mida ta haldab ning millele tal on ligipääs kuni lapse täisealiseks saamiseni. Lapsevanemal, kes loob lapsele konto, on õigus anda kontole ligipääs ka teisele lapsevanemale.

3.2.2 Ülesannete üleslaadimine

Kõik kõnehäired ei väljendu näost näkku teraapia vältel. Näiteks uues situatsioonis kogelus ei vallandu või valikulise mutismi tõttu ei saa laps võõras keskkonnas kõneleda. Samuti pole võimalik kõiki olulisi aspekte kabinetis hinnata. Seetõttu on tervikpildi saamiseks, probleemide ulatuse hindamiseks ning võimalikult täpse diagnostika tegemiseks vajalikud erinevate hindamismeetodite kasutamine. Sellest lähtuvalt võib tekkida vajadus, et patsient filmiks vastavalt spetsialisti poolt antud kavale erinevaid struktureeritud tegevusi kodus. Patsient saab sooritusel rakenduses üles laadida. Kui tegemist on lapse kontoga, siis on videole võimalik ligi pääseda teisel lapsevanemal tingimusel, et talle on see õigus antud.

3.2.3 Tagasiside ülesannetele

Pärast patsiendi poolt filmitud tegevuste üleslaadimist saab spetsialist teavituse, et videot on temaga jagatud. Seejärel on spetsialistil võimalik rakendusse sisse logides faili vaadata ja kommenteerida. Tagasiside kohta saab patsient teavituse ning võib soovi korral samuti kommentaaridele vastata. Teavitused saadetakse kasutajale meilile. Logopeedil on võimalik komplitseeritumate juhtumite puhul konsiiliumiks videot jagada teiste spetsialistidega.

3.2.4 Spetsialisti vaade rakenduses

Logopeedidel peab olema mugav ja hea ülevaade nendega jagatud failidest. Selleks, et logopeedi tööd efektiivsemaks muuta, on logopeedidel võimalik faile sildistada. Sildid saab iga logopeed endale ise luua. Märgistamine aitab logopeedil sarnase probleemiga juhtumid kiiresti rakendusest üles leida. Peale selle on logopeedil võimalik faile filtreerida ja sorteerida, näiteks uudsuse, kestuse, patsiendi nime alusel jne. Viimaks on logopeedil võimalik vaadata ühe patsiendi tervikprofiili, kus on järjestatud kõik üles laaditud videod, soovitused ja kommentaarid.

3.2.5 Harjutusvideote vaatamine

Rakenduse peamine eesmärk on hoiustada videoid, kuid rakenduses peab olema ka mugav viis videote vaatamiseks ja võrdlemiseks. Selle tagab võimalus vaadata videoid veebilehitsejast. Videomängija peab võimaldama videot mängida kiiremini ja aeglasemalt. Lisaks on oluline helifailide mängimise võimalus sest osa üles laetud failidest on audio kujul.

3.2.6 Videokõne võimalus

Rakendus võimaldab spetsialistidel läbi viia kaugteraapiat. Sisse loginud kasutajatel avaneb võimalus liituda kõnega. Logopeedidel on võimalus teraapiat salvestada, et pärast vajaduse korral oleks võimalik teraapiat uuesti vaadata. Vahel on salvestuse olemasolu oluline, et tõestada teraapia toimumist näiteks Eesti Haigekassale.

3.2.7 Mitme keele tugi

Platvormi hakatakse kasutama esmalt Eestis ning peamiselt kõnelevad patsiendid eesti keelt, kuid vajalik on valmisolek ka vene ja inglise keelt kõnelevate kasutajate jaoks. Sellest lähtuvalt tuleb tarkvara disainida nii, et see oleks kohaldatav erinevatesse keeltesse, ilma et tuleks teha täiendavaid muudatusi äriloogikas [5].

4 Analüüs

Käesoleva peatüki eesmärk on leida erinevatele nõuetele parim võimalik lahendus. Selleks on esmalt vaja kirjeldada, milliseid nõudeid peab sobiv lahendus täitma. Lisaks rakendusele esitatud nõuetele tuuakse välja ka lisapiirangud, mis kehtivad antud projekti raames.

Igas alapeatükis on välja toodud konkreetne nõue, mida on rakenduses vaja. Autor kirjeldab erinevaid võimalusi nõude realiseerimiseks. Tugevusi ja nõrkusi analüüsides leitakse kõige otstarbekam lahendus, võttes arvesse rakendusele esitatud piiranguid.

4.1 Rakenduse turvalisus

Antud lõputöö raames ei käsitleta kõiki turvalisuse aspekte nagu käideldavus ja terviklus. Täieliku turvaanalüüsi tegemiseks konsulteeritakse tulevikus oma ala eksperdiga, kes annab lõpliku hinnangu rakenduse turvalisusele. Selle peatüki raames käsitletakse, millised turvameetmed on kasutusel andmete varundamiseks ja andmete konfidentsiaalsuse tagamiseks.

4.1.1 Turvameetmed

Rakenduse arendamisel lähtutakse OWASP *Top Ten* dokumendist, mis käsitleb kõige kriitilisemaid veebirakenduste turvariske [6]. Seda peetakse silmas kogu arendusprotsessi vältel, et vältida levinud turvaauke.

4.1.1.1 Võtmete hoiustamine

Rakendusliides suhtleb andmebaasiga ja mitme välise teenusega, mis nõuavad võtmete või salasõna abil autentimist. Rakenduse võtmete hoiustamise võimalusi on mitmeid: koodivaramus, serveri keskkonnamuutujates või kasutatakse välist teenusepakkujat.

Koodivaramus olevad võtmed tuleks kindlasti üles laadida krüpteeritud kujul, kasutades selleks näiteteks git-crypt teenust, mis krüpteerib kasutaja poolt määratud failid koodivaramusse üleslaadimisel ning dekrüpteerib need allalaadimisel.

Teine variant on salvestada võtmed serveri keskkonnamuutujatesse, mis ei nõua täiendava teenuse kasutamist, kuid siis peab keegi manuaalselt serverisse sisse logima ja keskkonnamuutujad määrama.

Kolmas variant on kasutada välist teenust võtmete hoiustamiseks nagu näiteks HashiCorp Vault. Sarnase teenuse kasutamine võimaldab jälgida, millal on kasutatud erinevaid võtmeid ning võtmete hoiustamine toimub ühes kohas. Miinuseks on asjaolu, et teiste võtmete pärimiseks kasutatava võtme lekkimisel saab ründaja ligipääsu kõikidele võtmetele.

Rakendus evitatakse kliiniku serverisse, mida haldab süsteemiadministraator. Võtmete hoidmine keskkonnamuutujatena ei lisa välist sõltuvust ning on lihtsasti hallatav süsteemiadministraatori poolt. Seega on antud projekti raames otstarbekas kasutada keskkonnamuutujaid, mis antakse rakendusele kaasa käivitamise hetkel. Täiendava turvalisuse tagamiseks vahetatakse salajasi võtmeid ettemääratud aja tagant.

4.1.1.2 SQL-süstimine ja failide üleslaadimine

Patsientidel on õigus üles laadida video- ja helifaile, mis peavad olema valideeritud. Vastasel korral võib juhtuda, et üles laaditud fail on pahatahtlik skript. Lisaks failide edastamisele on rakenduses võimalus ka tagasisidet anda, mis on tekstiväli. Selleks, et vältida SQL-süstimise ohtu, kasutatakse kindlate parameetritega päringuid.

4.1.1.3 Eriliiki isikuandmete hoiustamine

Rakenduses hoiustatakse nii patsientide üles laaditud faile kui ka videotele antud tagasisidet. Nii faile kui ka tagasisidet võib liigitada eriliiki isikuandmetena, sest videotel on nähtav patsiendi tervislik seisund ja tagasiside võib seda samuti sisaldada [7].

Enne rakenduse kasutusele võtmist tehakse kasutajatele ülevaade rakendusest ning selle võimalustest. Spetsialiste juhendatakse hoiduma terviseandmeid sisaldavate kommentaaride andmisest. Sellegipoolest, kuna tegu on tekstiväljaga, võib see paratamatult põhjustada olukorra, kus spetsialisti tähelepanematuses sisaldab väli patsiendi terviseandmeid.

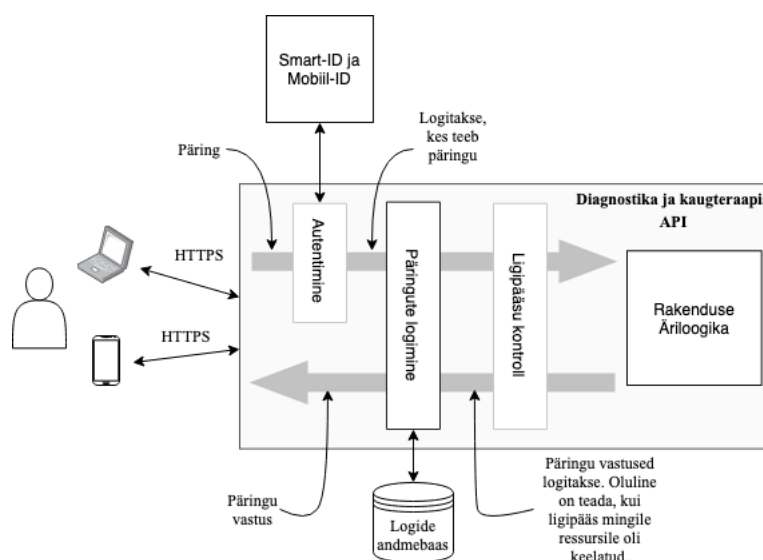
Et sellise olukorra tekkimist minimeerida, võetakse kasutusele järgmised meetmed: tagasiside vormi nägemiseks tehakse täiendav päring ning tekstivälja juures on kiri, et

tagasiside ei tohi sisaldada eriliiki isikuandmeid. Täiendava päringu tegemine tagasiside nägemiseks tagab selle, et on selge ülevaade, kes ja millal on seda vaadanud.

Videofailide salvestamisprotsessi kirjeldatakse põhjalikumalt peatükis 4.2. Sellegipoolest on oluline siin välja tuua, et kuna tegu on eriliiki isikuandmetega, siis võetakse kasutusele sarnased meetmed, mis tagasiside väljadega.

4.1.1.4 Monitoorimine ja logimine

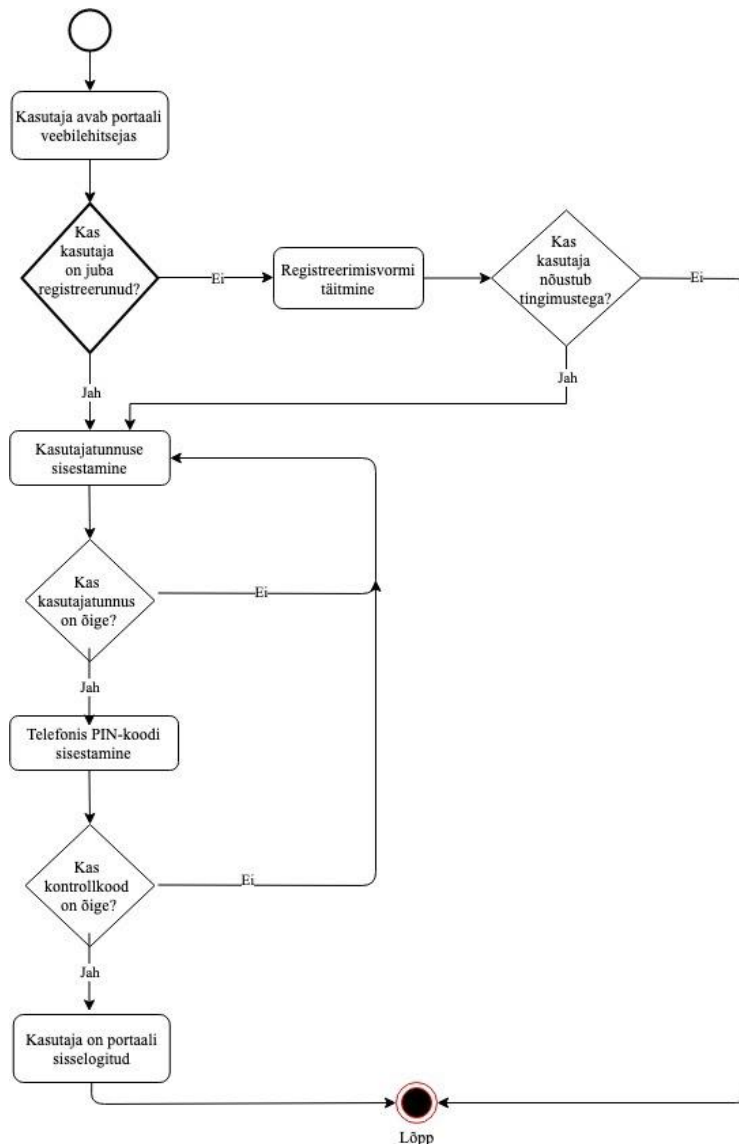
Kõiki päringuid logitakse ja salvestatakse (Joonis 1). Päringud eriliiki isikuandmetele salvestatakse koos kasutaja unikaalse identifikaatoriga, et patsiendil oleks selge ülevaade, kes on tema andmeid vaadanud. Lisaks päringute salvestamisele monitooritakse rakendust. Erinevate vigade avastamiseks kasutatakse Sentry tarkvara, mille abil on kiiresti võimalik rakenduses olevatele vigadele reageerida ning need parandada [8].



Joonis 1. Päringute logimine.

4.1.2 Autentimine ja autoriseerimine rakenduses

Nõuetest lähtuvalt toimub autentimine Smart-ID ja Mobiil-ID abil (Joonis 2). See lisab rakendusele täiendava turvaaspekti, sest kasutaja peab sisselogimiseks kasutama oma mobiili ja sisestama PIN-koodi. Seega on ründajal konto ülevõtmiseks vaja ligipääsu kasutaja telefonile ja peab teadma tema PIN-koodi.



Joonis 2. Autentimise voog.

Rakenduses kasutatakse rollipõhist ligipääsu kontrolli. Esialgu on kolm rolli: patsient, logopeed ja administraator. Vaikimisi määratakse kõigile patsiendi roll. Administraatori roll antakse kliiniku süsteemiadministraatorile, kellel on õigus muuta ja anda kasutajatele täiendavaid õigusi.

4.1.2.1 Patsientide kontod

Suur osa kliiniku patsientidest on alaealised. Neil ei pruugi olla Mobiil-ID või Smart-ID kontot. Sellest tulenevalt peab lapsevanemal olema võimalus avada lapsele konto. Eraldi rolli laste konto jaoks pole, kuid andmebaasis on viide, et tegu on alaealise kontoga ning lapsevanem saab sellele ligi. Lisaks on lapsevanemal õigus anda ligipääs ka teisele lapsevanemale.

Lastel on täisealiseks saamise hetkel õigus lastevanemate ligipääsu kontole piirata. Patsiendil on õigus enda konto kustutada. Teraapia vältel on oluline, et videod oleksid alles ja neid ei kustutataks. Sellegipoolest, kui patsiendil on soov teraapia vältel videoid kustutada, siis on tal seda võimalik teha, esitades soovi administraatorile. Kümme aastat pärast tervishoiuteenuse osutamist kustutatakse patsiendiga seotud failid automaatselt [9].

4.2 Failide salvestamine

Videofailide salvestamiseks on võimalik kasutada olemasoleva veebiserveriga ühendatud ketast, kus on hetkel vaba ruumi 100 gigabaiti. Teine variant oleks videofailid salvestada kuhugi mujale, kasutades selleks näiteks objekti ladu (*object storage*) lahendust. Hetkel on kliinikul ligikaudu 50 gigabaiti erinevaid faile, mis on salvestatud kettale. Kliinik soovib tulevikus hakata näidisvideoid üles laadima, mistõttu tuleb tulevikus osta juurde täiendav ketas või kasutada kolmandat osapoolt täiendava andmemahu rahuldamiseks.

Täiendava turvalisuse ja mastabeeritavus saavutamiseks kasutatakse antud projekti raames videote ja tulevikus ka näidiste salvestamiseks kolmandat osapoolt. Sellise lähenemise puhul ei pea muretsema, millal olemasolev ketas täis saab. Teenusepakkujate valimisel tuleb lähtuda, et faile hoitaks Euroopa Liidus [7].

2020. aasta seisuga on infrastruktuuri ja pilve teenuste turuliirid Amazon, Microsoft ja Google [10]. Lisaks turuliidrite pakkumistele võetakse võrdlusesse ka Eestis tegutsev pilveteenust pakkuv ettevõtte Pilvio, sest nende andmekeskused on Eestis. Viimaks lisatakse võrdlusesse DigitalOcean pakkumine, mis pakub samuti objekti lao võimalust Euroopas. Võrdlusest jäeti välja Eestis tuntud teenusepakkujad Zone ja Wavecom, sest nad ei paku võrreldavat teenust (Tabel 1).

Tabel 1. Pilveteenuse pakkujate võrdlus.

Teenuse-pakkuja	Andmekeskuse asukoht	Hind	Liikluse tariif
Amazon Web Service	Frankfurt, Saksamaa	0,0245\$ GB/kuus	Lisandub päringute eest tehtud tasu ja liikluse tasu.

Teenuse-pakkuja	Andmekeskuse asukoht	Hind	Liikluse tariif
Google Cloud Platform	Hamina, Soome	0,020\$ GB/kuus	Lisandub päringute eest tehtud tasu ja liikluse tasu.
Azure	Amsterdam, Holland	0,16445€ GB/kuus	Lisandub päringute eest tehtud tasu ja liikluse tasu.
Pilvio	Jõhvi, Eesti	0,025€ GB/kuus	Kuni 15TB kuus tasuta, täiendav maht pole täpsustatud.
DigitalOcean	Frankfurt, Saksamaa	5\$/ 250 GB kuus. Täiendav maht: 0,2\$/GB	Kuni 15TB kuus tasuta, täiendav maht 0,01\$/GB.

Kõikidel välja toodud teenuse pakkujatel on olemas tarkvaraarenduse komplekt (SDK), mis teeb rakendusliidesega suhtluse lihtsaks ja mugavaks. Pilvio ja DigitalOcean teenused ühilduvad Amazoni tarkvaraarenduse komplektiga.

Kõik suuremad teenusepakkujad Amazon, Google ja Microsoft pakuvad küll soodsat failide hoiustamist ja maksuma peab ainult kasutatava mahu eest, kuid kõigile lisandub raskesti ennustatav liikluse ja päringute tariif, mis võib algselt soodsana tunduva teenuse muuta palju kallimaks.

DigitalOcean pakub vahepealset lahendust, kus puuduvad teenustasud tehtud päringute eest ja esimesed 15TB liiklust kuus on tasuta. Miinuseks on asjaolu, et peab maksuma ruumi eest, mida ei kasutata, kui failide maht on väiksem kui 250 GB.

Pilvio on ainuke teenusepakkuja, kus makstakse üksnes kasutatud mahu eest ja kuni 15TB kuu puhul puudub liikluse tariif. Lisaks on teenusepakkuja serverid Eestis, mistõttu on latentsus väiksem.

Antud rakenduse videofailide salvestamiseks kasutatakse Pilvio platvormi, mis on kõige soodsam, ning ühilduv Amazon S3 tarkvaraarenduse komplektiga. Lisaks on Pilvio Eestis ning latentsus on võrreldavate teenusepakkujate seas kõige väiksem.

4.3 Videomängija lahenduse analüüs

Rakendus peab võimaldama patsientidel ja logopeedidel vaadata üleslaetud videoid. Faile peab olema võimalik logopeedidel märgendada, et videod võimalikult kiiresti üles leida. Veebirakendus peab võimaldama vaadata videoid enamlevinud veebilehitsejatest ja videomängija peab kohalduma erinevate ekraanisuurustega.

4.3.1 Olemasolevad lahendused turul

HTML5 standardiga avaldati video märgend (*tag*), mis võimaldab videoid veebilehitsejas lisada ilma väliste teekide abita [11]. Enne video märgendi olemasolu oli videote mängimiseks enimlevinud Flash Player, mis tänaseks on aegunud [12]. Seega uuritakse ainult neid lahendusi, mis teostavad HTML5 standardit.

Nõuded, mida videomängija peab võimaldama:

- Videot saab peatada ja käivitada
- Video kiirust saab muuta
- Video heli saab muuta
- Videomängija välimus peab ühtima ettevõtte brändi värvidega

Uuritakse avatud lähtekoodiga teeke, mis eelnevalt nimetatud nõuded täidaks ning arendustöid kiirendaksid. Tasulised videomängijad jäetakse analüüsist algselt välja. Kui vabavaralised lahendused ei täida esitatud nõudeid, siis analüüsitakse tasulisi variante.

Autor leidis järgmised teegid, mis täidavad videomängijale esitatud nõuded: VideoJs, Plyr, Mediaelement ja GLightbox. Antud rakenduses kasutatakse VideoJs teeki, mis on mainitud teekidest kõige populaarsem, arvestades GitHubis olevaid tärne. Lisaks on VideoJs teegil olemas pistikprogrammide süsteem, mille abil on võimalik erinevaid võimalusi lihtsasti lisada. Samuti on VideoJs antud teekide seast kõige põhjalikumalt dokumenteeritud.

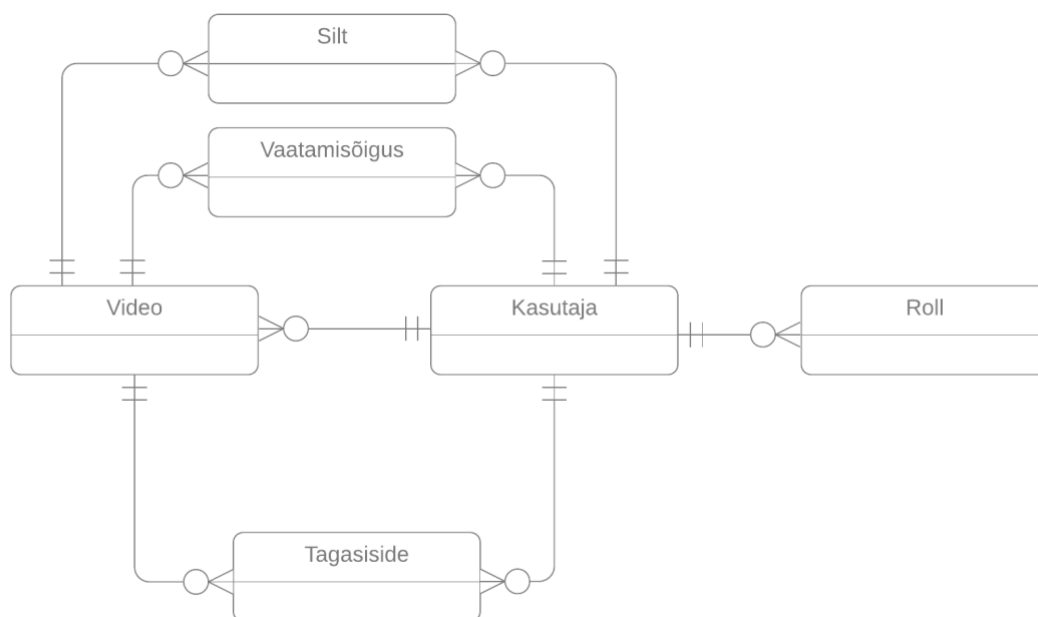
4.3.2 Videomängija serveripoolne lahendus

Peale videote vaatamise võimaluse peavad videomängijal olema järgmised võimalused:

- Logopeedid saavad faile sildistada
- Logopeedid näevad patsiendi poolt üles laaditud faile
- Logopeedid saavad failidele kommentaare lisada

- Patsiendid saavad tagasisidele vastata

Selleks, et eelnevalt mainitud nõuded täita, on vaja luua vastav andmemudel. Joonisel 3 on välja toodud andmemudel, mis võimaldab täita kõik videomängijale esitatud ärinõuded.



Joonis 3. Videomängija andmemudel.

4.3.3 Teavituste saatmine

Selleks, et patsiendid saaksid võimalikult kiiresti tagasisidet videotele, siis peab rakenduses olema ka teavituste süsteem. Esialgu on nõue, et teavitused peavad patsiendile ja logopeedile e-kirja teel tulema. Kliinikul on olemas Twilio konto, mille kaudu on varasemalt patsientidele teavitusi saadetud. Sellest tulenevalt kasutatakse antud projekti raames e-kirjade saatmiseks Twilio Sendgrid teenust. Klient on avaldanud soovi tulevikus teateid saata SMS sõnumite teel, mida on samuti võimalik Twilio kaudu teha.

4.4 Videokõne võimalus

Rakenduses peab olema võimalus teha videokõnesid, et pakkuda kaugteraapiat. Videokõne lahendus peab täitma järgmisi nõudeid:

- Videokõne tegemiseks piisab rakenduses konto omamisest
- Videokõne tegemiseks ei pea alla laadima täiendavaid rakendusi

- Videokõne on võimalik logopeedil salvestada
- Videokõnes osalejad saavad enda kaamerat sisse ja välja lülitada
- Videokõnes osalejad saavad enda mikrofoni sisse ja välja lülitada
- Videokõnes osalejad saavad enda ekraani jagada
- Salvestatud videoid on võimalik logopeedil rakendusest vaadata

Olemasolevad valmislahendused – Skype, Microsoft Teams, Zoom jpt ei sobi. Kõik lahendused täidavad küll enamik nõuetest, kuid neid pole võimalik integreerida rakendusega. Osa nõuab täiendava tarkvara allalaadimist, puudub selge kontroll, kellel on õigus videot salvestada ja salvestatud videod pole koheselt loodavas rakenduses olemas.

Sellest tulenevalt tuleb luua kohandatud lahendus. Arvestades, et esialgne rakendus on veebipõhine, siis on paslik kasutada WebRTC tehnoloogiat videokõne võimaluse lisamiseks. See on olemas enim levinud veebilehitsetajates ning iOS ja Android seadmetel on võimalik seda kasutada teekide vahendusel [13].

Vabavaralised lahendused, mis täidavad videokõne nõuded, on BigBlueButton ja Jitsi Meet. BigBlueButtoni eeliseks võib pidada õppetöö tegemiseks täiendavaid võimalusi tahvli olemasolu ja märkmete jagamine. Vabavaraliste lahenduste kasutamise miinusena võib välja tuua, et tarkvara tuleb ise hallata. Teiseks on vaja riistvara, millel tarkvara jooksutada. Konsulterides kliendiga selgus, et hetkel puudub kliinikul ressurss, et tarkvara poolt esitatud riistvara nõudeid täita.

Turul on olemas tasulised lahendused, mis kasutavad WebRTC tehnoloogiat. Amazon Chime, Twilio Video jpt on ehitanud mitmeid võimalusi nagu videote salvestamine, ekraani jagamine ja taustamüra vähendamine. Tasulise tarkvara kasutamine suurendab rakenduse ülalpidamiskulusid, kuid kiirendab arendusprotsessi. Arvestades, et nõudlus kaugteraapia teenusele pole teada, kasutatakse algselt tasulist lahendust. Nõudluse olemasolu korral otsustatakse tulevikus uuesti, kas on otstarbekas realiseerida võimalus ise või jätkata kolmanda osapoole kasutamist.

Videokõne võimaldamiseks kasutatakse rakenduses Twilio Video teenust. See pakub kahe osaleja vahel kõnesid tasuta, mis on peamine kasutusjuhtum. Antud võimaluse vajadust rakenduses on kõige soodsam katsetada Twilio poolt pakutava teenusega.

4.5 Tehnoloogiate valik

Rakenduse arendamiseks vajalike tööriistade valikul lähtutakse äripoolsetest nõuetest. Nõuded, mida rakendus peab täitma on järgmised:

- Videokõne patsiendi ja logopeedi vahel
- Videote üleslaadimise võimalus
- Rakendus peab töötama kliiniku serveris
- Rakendus peab olema adaptiivne (töötama nii mobiilis kui ka arvutis)
- Üleslaaditud videoid peab olema mugav veebilehitsejast vaadata

4.5.1 Serveripoolne rakendus

Üks nõue oli, et tulevikus peab olema lihtne lisada uusi võimalusi. Lisaks tuleb tulevikus arvestada, et rakendusele on oodata täiendavaid muudatusi ja uuendusi, mistõttu jäävad vaatluse alla ainult tüübitud programmeerimiskeeled. Populaarsemad tüübitud keeled on TypeScript, C#, Java.

Java suureks plussiks on Spring raamistik, mida on arendatud alates 2003. aastast ja arendatakse väga aktiivselt tänaseni [14]. Java kasutusele võtmisel on tulevikus võimalik rakenduses kasutada ka muid keeli, mis on kompileeritavad Java baitkoodiks. Lisaks on Spring projektil moodul Spring Security, kus on väga palju olulisi turvalisuse aspekte arendaja jaoks ette tehtud. Viimaks räägib Java valiku kasuks ka see, et SK Solutions, kes pakub Smart-ID ja Mobiil-ID lahendust, on arendanud valmis Mobiil-ID ja Smart-ID Java teegi [15].

TypeScripti kasutamise suureks eeliseks oleks võimalus nii kliendi- kui ka serveripoolne rakendus kirjutada samas keeles. Miinusena võib välja tuua, et TypeScriptil pole selget raamistikku, kus oleks turvalisus kohe lahendatud. Seega tuleb arvestada täiendava ajakuluga, mis on seotud rakenduse turvalahenduse teostamisega. Lisaks on TypeScripti arendus serveripoolsete rakenduste puhul Eestis vähemlevinud kui Java kasutamine.

C# kasuks räägib kindlasti see, et selle taga on üks maailma suurimaid ettevõtteid – Microsoft. C# keeles on korralik raamistik .NET Core, mis jookseb nii Linuxi kui Windowsi operatsioonisüsteemil. Puuduseks võib lugeda asjaolu, et kliiniku varasemad koostööpartnerid ei kasuta C# keelt arendamiseks, vaid Javat ja PHP-d.

Selle projekti raames kasutatakse Java programmeerimiskeelt ja Spring raamistikku. Põhjuseks on varasemate koostööpartnerite oskus hallata ja arendada Java rakendusi. Teiseks on suur ajaline võit Spring Security raamistiku kasutamisel ja kliendi teegi olemasolu Mobiil-ID ja Smart-ID integreerimisel.

4.5.2 Esilehe rakendus

Hoides klientrakendust ja serveripoolset rakendust eraldi, siis on tulevikus lisaks veebipõhisele lahendusele lihtsam juurde luua mobiilipõhised lahendused, sest tagarakendus on ehitatud selliselt, et päringule vastatakse JSONiga. Lisaks sellele saab rakendust osaliselt kasutada ka ilma interneti ühenduseta, kui arendada rakendus progressiivse veebirakendusena (*Progressive Web App*) [16].

Veebipoolne rakendus lahendatakse üheleherakendusena, edaspidi SPA, mis hoiab kliendi- ja serveripoolse koodi lahus. Populaarsemad SPA raamistikud on Vue.js, AngularJS ja React [17]. Raamistik peab võimaldama paindlikkust, et lisada videomängija ja videokõne võimalus. Kõik raamistikud on laialdaselt kasutusel ja täidavad rakendusele esitatud nõuded.

Teegi valikul tegi autor otsuse lähtuvalt kliendi soovist tulevikus luua ka mobiilirakendus. React on võimalik kasutusele võtta nii veebi- kui ka mobiilirakendusi ehitades. Selle tulemusel on võimalik osa koodi taaskasutada ka mobiilirakenduses, muutes mobiilirakenduse arendamise tulevikus kiiremaks ja soodsamaks.

4.5.3 Andmebaasi haldussüsteemi valik

Olemasoleva rakenduse andmed on salvestatud MySQL andmebaasi. Uus rakendus peab võimaldama olemasolevatel kasutajatel uude rakendusse sisse logida ja üleslaaditud videotele ligi pääseda.

Rakenduses hoitavad andmed on isiku poolt üles laaditud videod ja isikuandmed tema tuvastamiseks. Isikuandmeid on mugavalt võimalik hoiustada relatsioonilises andmebaasis ja kasutaja poolt üles laaditud videote asukoha saab salvestada viitena. Rakendus evitatakse kliiniku serverisse, kus on olemas MySQL andmebaas.

Andmebaasid, mis rakenduse nõuded täidaks ja oleks vabavaralised on MySQL ja PostgreSQL. Antud rakenduse raames on sobilik kasutada olemasolevat andmebaasi, mis

täidab rakenduses esitatud nõuded ja ei nõua täiendava andmebaasi haldussüsteemi paigaldamist. Viimaks on kliinikul olemas süsteemiadministraator, kes seda haldab.

4.6 Arendusprotsess

Projekti raames lähtutakse välearendus metoodikast. See tähendab, et iga kahe nädala tagant saadakse kliendiga kokku ja näidatakse, mida on vahepeal tehtud. Iga sprindi lõpus on valmis midagi, mis loob kliendile väärtust ning mille saab kohe kasutusele võtta [18]. Koodi hoitakse keskses koodivaramus ning iga uue funktsionaalsuse lisamisel luuakse uus haru (*git branch*).

4.6.1 Pidev integratsioon

Selleks, et pakkuda kliendile võimalust olla arendusprotsessiga kursis ja tulevikus uute võimaluste lisamine oleks võimalikult kiire ja lihtne, kasutatakse antud projektis pidevat integratsiooni (CI). Selline lähenemine võimaldab tulevikus ka uutel arendajatel, kes projektiga liituvad, kiiremini uusi võimalusi platvormile arendada.

Kliendi- ja serveripoolse rakenduse evitamiseks on vaja uute muutuste korral ehitada valmis uus versioon rakendusest. Seejärel tuleb uuendused automaatselt kliiniku veebiserverisse üles laadida. Selleks, et kliiniku serverisse poleks vaja erinevaid rakenduse sõltuvusi alla laadida, kasutatakse rakenduse ehitamiseks ja käivitamiseks kontaineriseerimise tehnoloogiat Docker.

Serverisse tuleb paigaldada rakenduse käivitamiseks ainult Dockeriga seonduv tarkvara [19]. Teiseks annab see kindluse, et rakendus töötab serveris, sest rakenduse konfiguratsioon ja keskkond on sama nii lokaalselt kui serveris. Tulevikus kui tekib vajadus serveri vahetuseks, siis on võimalik lihtsa vaevaga teenus migreerida mujale [20].

Turul on palju erinevaid pideva integratsiooni tööriistu – Jenkins, Buildkite, Gitlab CI/CD. Antud projekti raames kasutatakse koodi hoiustamiseks GitHub platvormi. Antud projektis on sobilik kasutada GitHub Actions teenust, sest lähtekoodi hoiustatakse seal ning see ei lisa täiendavat kulu. Tööriist toetab kasutajaliidese ja serveripoolse rakenduse ehitamist ning tõmmiste (docker image) loomist. Lisaks on võimalik tõmmised GitHub kontainerite registrisse üles laadida [21].

Viimaks, et ehitatud rakendus jookseks toodangukeskkonnas, tuleb rakendus serveris käivitada. Selle automatiseerimiseks kirjutatakse skript, mis logib kliiniku veebiserverisse, laeb alla kõige uuema tõmmise, peatab hetkel käimasoleva rakenduse ning loob uue konteineri. Probleemide esinemise korral on võimalik tagasi pöörduda varasema tõmmise juurde ja luua ning käivitada konteiner selle põhjal.

Selline lähenemine on sobilik antud kontekstis, kus rakendusel on piiratud arv kasutajaid. Tulevikus, kui tarkvara kasutab mitu kliinikut ja on oluline, et rakendus oleks pidevalt kättesaadav, on võimalik skripti täiendada ning muuta evitamise protsessi peenenemaks ja lisada täiendavaid samme, kasutades selleks näiteks *Canary release* tehnikat [22].

5 Praktiline lahendus

Peatükis 4 analüüsis autor erinevaid tehnilisi võimalusi nõuete teostamiseks ning valis välja kõige sobivamad. Selleks, et tehnilisi valikuid valideerida, arendatakse rakendus, kus kasutatakse analüüsi käigus välja toodud tehnoloogiaid. Antud peatükk annab ülevaate, kuidas olulisemad nõuded teostati.

5.1 Rakendusliidese kirjeldamine ja kasutamine

Projekti raames alustati kõigepealt tagarakenduse arendamisega. Kirjutati valmis rakendusliidese lõpp-punktid ning äri loogika. Esirakenduses tehtavad päringud ja päringute lõpp-punktid ning mudelid genereeriti, kasutades selleks OpenAPI spetsifikatsiooni [23].

Seega kirjutatakse kõigepealt serveri poolel uus lõpp-punkt ning seejärel genereeritakse kirjutatud koodist spetsifikatsioon. Selle alusel on võimalik luua klient, kus on kirjas lõpp-punktid, kuhu tuleb päring saata. Lisaks on kirjas, milline on sisend, mida oodatakse ning millised on võimalikud väljundid. Seeläbi on nii kliendi- kui serveripoolne rakendus sünkronis, sest esirakendusest saadatud mudelid on genereeritud serveri koodi põhjal.

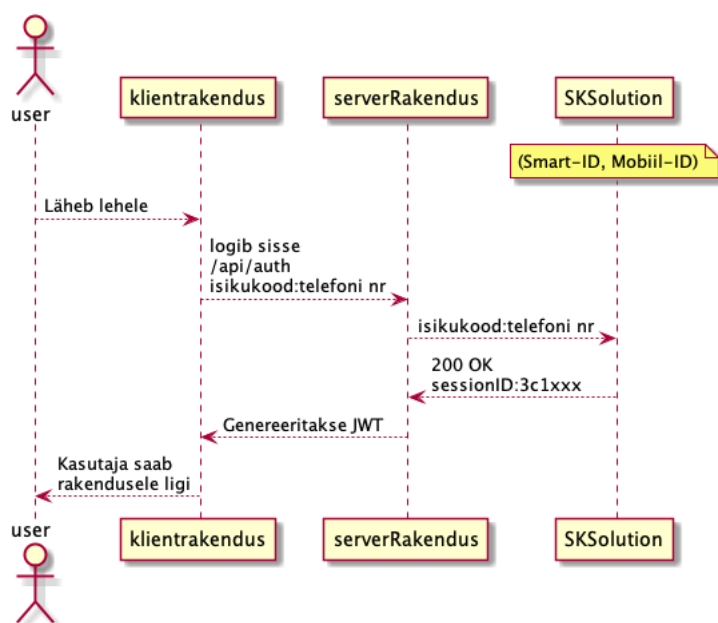
5.1.1 Probleem OpenAPI kasutamisega

Algne lähenemine kirjutada kõigepealt valmis rakendusliides ning seejärel genereerida klient esirakendusele polnud praktikas kõige otstarbekam. Pärast kliendiga kokkusaamist soovis klient mitut täiendust ning muutusi klientrakenduses. See tähendas omakorda seda, et tuli teha täiendavad lõpp-punktid rakendusliideses ning genereerida uus kood.

Esialgu hoidis koodi genereerimine aega kokku, sest klientrakenduses olid mudelid ja lõpp-punktid määratud, kuid pidevate muutuste korral rakendusliideses tähendas kogu protsessi kordamist, mis vahel oleks olnud võimalik muuta paari rea muutmisega klientrakenduses. Töö autor otsustas esialgu genereeritud koodi alles jätta, kuid täiendavaid muutusi teha klientrakenduses käsitsi.

5.2 Autentimine rakenduses

Peatükis 4.5.1 leiti, et parim raamistik serveripoolse autentimise ja autoriseerimise teostamiseks on Spring Security. Autentimisprotsessi parema ülevaate saamiseks on Joonisel 4 toodud välja, kuidas toimub päring ning milline on tulemus, kui autentimine õnnestub.



Joonis 4. Järgnevuskeem – kasutaja autentimine.

Kasutaja peab rakenduse kasutamiseks sisestama oma isikukoodi ning telefoni numbri. Seejärel saadetakse päring tagarakendusse, mis omakorda saadab päringu SK ID Solutions teenusele. Eduka sisselogimise korral genereeritakse luba, kus on kirjas kasutaja õigused ning loa kehtivusaeg. Ebaõnnestumise korral kuvatakse kasutajale vastav veateade ning ümbersuunamist ei toimu.

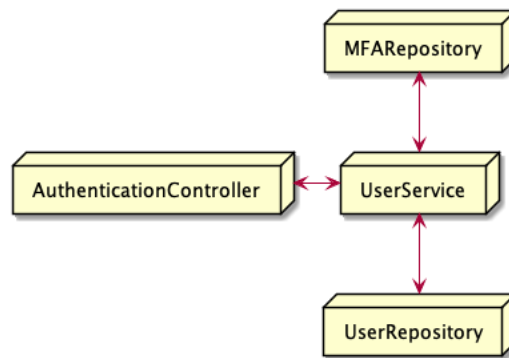
Klientrakenduses tuleb luba salvestada, et see iga päringu päisesse lisada. Selle salvestamiseks kasutatakse React Contexti, kuhu on võimalik salvestada andmed selliselt, et kõik komponendid saavad sellele ligi [24]. Kasutajainfo salvestamine globaalselt on oluline, et kontrollida, kas kasutajal on õigus ligi pääseda kindlale komponendile.

5.2.1 Autentimine toodangu- ja arenduskeskkonnas

Peatükis 4.1.2 toodi välja, et autentimine toimub Smart-ID ja Mobiil-ID abil. Selline lähenemine on mõistlik toodangukeskkonnas, kuid rakenduse testimise faasis oleks mugav, kui saaks rakendust kasutada salasõna alusel. Selle võimaldamiseks kasutatakse

raamistiku poolt pakutavaid profile, kus on võimalik seadistada, kuidas toimub autentimine arendus- ja toodangukeskkonnas [25].

Selleks, et arenduskeskkonnas kasutataks salasõna abil autentimist, seadistatakse Spring Security raamistik niiviisi, et kasutajate haldamiskomponent muutub vastavalt keskkonnamuutujale, millist autentimissüsteemi parasjagu kasutatakse (Joonis 5).



Joonis 5. Autentimine rakenduses sõltuvalt seadistusele.

5.3 Failide salvestamine

Peatükis 4.2 tõi autor välja, et erinevate salvestiste hoiustamiseks kasutatakse Pilvio teenust. Failidega operatsioonid toimuvad tarkvaraarenduse komplekti kaudu, mille kasutamiseks on vaja see lisada sõltuvusena projekti. Komplektiga tuleb kaasa klass, kus on kaasas vastavad meetodid failide kustutamiseks, üles- ja allalaadimiseks.

Klassist, mis on kaasas tarkvaraarenduse komplektiga, luuakse uus objekt. Isendi loomine toimub ainult ühe korra ehk kasutatakse Singleton nimelist disaini mustrit [26]. Tarkvaraarenduse komplekti kasutatakse rakenduse andmekihis, kus on tehtud kohandusi vastavalt vajadusele.

Enne failide saatmist andmekihti valideeritakse, et üleslaetav fail pole pahatahtlik ja on etteantud formaadis. Kontroll toimub nii kliendi- kui serveri poolel, kus kontrollitakse faili laiendit ning suurust. Maksimaalne faili suurus võib olla 500 megabaiti. Lubatud faililaiendid on enim levinud heli- ja video meediatüübid [27].

Kasutajaliidese arendamisel kasutatakse Ant Design komponendi teeki. Seal on olemas üleslaadimise komponent [28]. Seda kohandatakse, et kontrollida, kas üleslaetav fail on heli- või videofail (Joonis 6).


```

beforeUpload: (file: RcFile) => {
  const ALLOWED_FILETYPES = [
    'video/mp4',
    'video/ogg',
    'video/webm',
    'video/quicktime',
    'video/x-msvideo',
    'audio/mpeg',
    'audio/L24',
    'audio/basic',
    'audio/x-aiff',
    'audio/ogg',
    'audio/vorbis',
  ]
  if (ALLOWED_FILETYPES.indexOf(file.type) === -1) {
    openNotificationWithIcon('error', t('errorMsg'),
`${file.name} ${t('fileNotAllowed')}` , 2)
    return Upload.LIST_IGNORE
  }
  setUserData(file)
  return false
},
maxCount: 1,
}

```

Joonis 6. Failide valideerimine brauseris.

Serveri poolel kontrollitakse üleslaetud faile. Meediatüübi kontrollimiseks eraldi teeki ei kasutata, sest täiendava sõltuvuse lisamine suurendab lõpliku rakenduse suurust ning ei tasu autori arvates ära. Seejärel, kui on kontrollitud faili sobivust nii kliendi kui serveri poolel, siis laetakse fail üles objekti lattu ja salvestatakse faili kohta käivad andmed andmebaasi.

5.4 Video- ja helifailide mängija lahendus

Peatükis 4.3.1 valitud teek Video.js võimaldab mängida nii video- kui ka helifaile [29]. Rakenduses ühtlustati videomängija välimus ettevõtte värvidega, kasutades selleks raamistiku poolt pakutud malli [30]. Lisati ka võimalus video- ja helifailide kiirust muuta. Faile on spetsialistil võimalik sildistada ning filtreerida (Lisa 2).

Rakenduses failide mängimiseks ei saadeta üles laaditud faile esirakendusele, vaid viidatakse sellele. Tänu sellele, et Pilvio teenus on ühilduv Amazon S3 tarkvaraarenduse komplektiga, siis on võimalik luua unikaalne URL, millele on võimalik kliendi poolel viidata [31]. Kui spetsialist või patsient avab video, siis tehakse päring serverisse, mis kontrollib, kas kasutajal on õigus antud ressursile ligi pääseda ning seejärel tehakse esirakendusele kättesaadavaks veebiaadress, mille abil on võimalik heli- ja videofaile brauseris mängida (Joonis 7).

```

public URL getUserVideo(User user, String title, Integer time) {
    var client = getS3Client();
    var timeValid = Date.from(LocalDate.now()
        .plusMinutes(time).atZone(ZoneId.systemDefault()).toInstant());
    return client.generatePresignedUrl(bucketName, getObjectPath(
        user.getUsername(), title), timeValid, HttpMethod.GET);
}

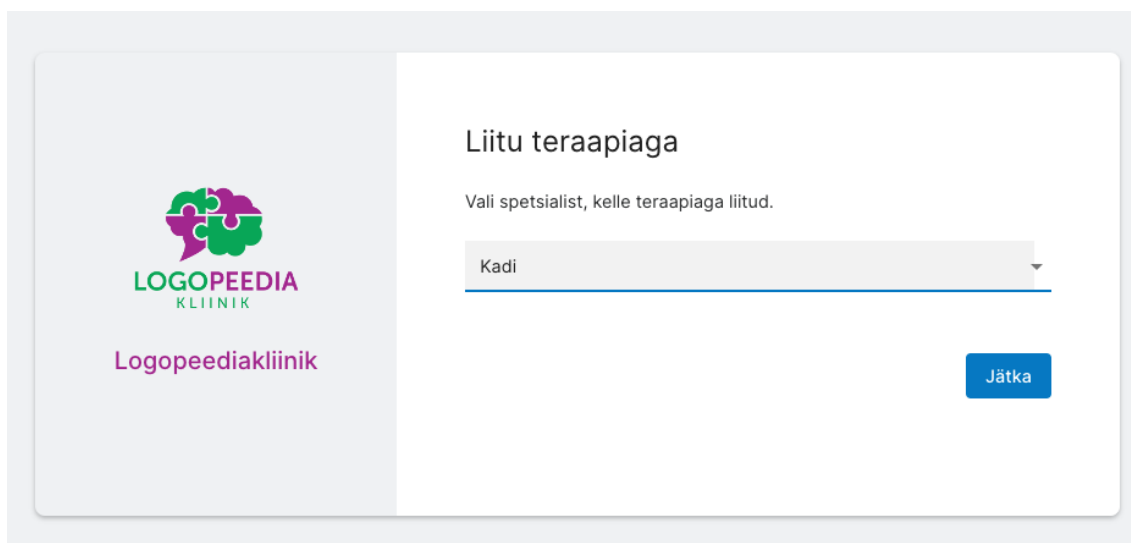
```

Joonis 7. Signeeritud veebiaadressi genereerimine tagarakenduses.

Serveri poolel päritakse andmeid, kasutades selleks DAO disaini mustrit. See abstrahereib kogu suhtluse failide salvestamisel ja pärimisel. Tulevikus on andmete salvestamise asukohta võimalik muuta ilma, et tuleks teha muutusi rakenduse äriloogikas [32].

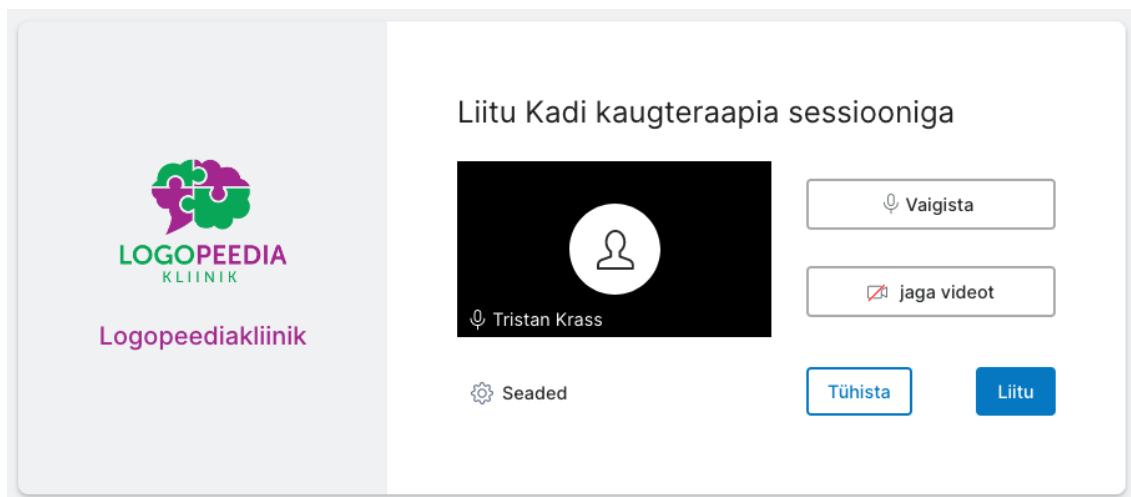
5.5 Kaugteraapia pakkumise võimalus

Peatükis 4.4 otsustati kasutada Twilio teenust kaugteraapia võimaldamiseks. Kaugteenuse pakkumine on lahendatud järgmiselt: igale tugispetsialistile on loodud üks videokanal, millega on patsientidel võimalik liituda [33]. Patsientidel on rakenduses ülevaade spetsialistidest kellega neil on võimalik liituda (Joonis 8).



Joonis 8. Kaugteraapiaga liitumine rakenduses.

Enne kõnega liitumist on võimalik kasutajal seadistada kaamerat, mikrofoni ning video kvaliteeti (Joonis 9). Spetsialistidel on võimalus kaugteraapiat salvestada (Lisa 3). Selleks kasutatakse Twilio Video teenust.



Joonis 9. Videokõne seadmistamise võimalus.

Salvestuste vaatamiseks on tagarakenduses loodud lõpp-punkt, mis suhtleb Twilio rakendusliidesele [34]. Tagarakendus pärib kõik spetsialistiga seotud videod ning teeb videod esirakendusele kättesaadavaks.

5.6 Mitme keele tugi

Antud projekti raames lisatakse tõlked esialgu ainult kliendipoolsele rakendusele. See tähendab, et tõlgitakse ära kasutajaliides. Selle saavutamiseks kasutatakse I18next teeki, mis võimaldab tõlkeid hoida failis ning seeläbi on lihtne kolmandatel osapooltel tõlkeid lisada või muuta [35].

Selleks, et teeki kasutada, tuleb luua konfiguratsiooni fail. Seal on kirjas, mis keeli antud rakendus tõlgib, kus asuvad tõlkefailid ning vajadusel ka täiendavate pistikprogrammide konfiguratsioon (Joonis 10).

```
import i18n from 'i18next'
import { initReactI18next } from 'react-i18next'
import Backend from 'i18next-http-backend'
import LanguageDetector from 'i18next-browser-languagedetector'
import 'react-i18next'
i18n
  .use(Backend)
  .use(LanguageDetector)
  .use(initReactI18next)
  .init({
    fallbackLng: 'et',
  })
export default i18n
```

Joonis 10. I18n konfiguratsioon.

Lisaks seadistatakse kasutajaliidese ehitamiseks kasutatav teek, et see toetaks eesti, vene ja inglise keelt [36]. Kasutaja eelistused salvestatakse kliendi veebilehitseja mälusse ning hoitakse globaalses olekus. Vaikimisi on rakenduse keeleks valitud eesti keel.

5.7 Pidev integratsioon ja evitamine

Uute võimaluste lisamiseks luuakse koodivaramus uus haru. Loodud haru mestitakse peamise haruga, kui uus võimalus on valmis arendatud ning kood läbib testid edukalt. Testide automaatseks käivitamiseks kasutatakse GitHub Actions teenust.

Peatükis 4.6.1 toodi välja, et rakenduse evitamine toimub automaatselt ning selleks kasutatakse GitHub Actions teenust ja Docker tehnoloogiat. Selle abil ehitatakse uus tõmmis, mis laaditakse üles GitHub Container Registry keskkonda. Sealt on võimalik tõmmis serverisse alla laadida ja selle põhjal konteiner ehitada. Viimaks tuleb ehitatud konteiner käivitada.

Antud projekti raames kasutatakse teenuste käivitamiseks ja peatamiseks docker compose tööriista. See võimaldab teenuseid käivitada ühe käsuga, mis teeb teenuste haldamise lihtsaks. Tööriista kasutamiseks tuleb luua YAML laiendiga fail, kus on kirjas sammud, mida tuleb teha teenuse käivitamiseks (Joonis 11).

```
version: "3.9"
services:
  backend:
    image: docker.pkg.github.com/logospeed/portal:latest
    ports:
      - "5000:8080"
    env_file:
      - .env
    restart: unless-stopped
  frontend:
    image: docker.pkg.github.com/logospeed/portal-client:latest
    ports:
      - "80:80"
    restart: unless-stopped
```

Joonis 11. Docker compose fail serveri ja kliendi käivitamiseks.

6 Tulemused

Töö alguses püstitati kaks eesmärki: uue jätkusuutliku rakenduse tehnilise analüüsi tegemine ja uue rakenduse loomine. Peatükis 3.1 analüüsiti olemasoleva rakenduse puudusi ning toodi välja miks pole olemasoleva rakenduse arhitektuur jätkusuulik.

Autori hinnangul on eesmärk: leida jätkusuutlik lahendus, täidetud. Teine eesmärk oli luua uus rakendus, mis on hetkel veel arendusfaasis. Rakendust on võimalik kasutada, kuid enne lõplikku väljalaskmist tuleb teha täiendav analüüs rakenduse turvalisuse kohta.

6.1 Tagasiside

Valminud lahenduse loomisel on töö autor pööranud tähelepanu nii kasutajamugavusele kui turvalisusele. Kliiniku juhatus ja spetsialistid tõstsid esile märgendite süsteemi, mis võimaldab kindla profiili järgi patsiente leida, et vajadusel andmeanalüüsi teostada. Teiseks toodi esile võimalust audio- ja videofaile mängida brauseris ning võimalust esitluskiirust muuta.

Spetsialistid tõid positiivsena välja ka patsiendi tervikprofiili nägemise võimaluse (Lisa 4). Kasutajamugavust tõstab videokõne tegemise võimalus samas rakenduses, kus on olemas patsientide kohta käivad andmed. Lisaks on patsientide otsimine ja filtreerimine palju mugavam kui eelmises rakenduses. Spetsialistid tõid välja, et tulevikus võiks tagasiside väljale olla võimalus lisada spetsialistil erinevaid faile.

Kokkuvõtvalt leiti, et loodud lahendus on arusaadav ja mugav kasutada. Teostatud lahendus muudab spetsialisti töö oluliselt mugavamaks. Lahendus on plaanis kliinikus kasutusele võtta ning edasi arendada.

6.2 Tulevikuplaanid

Antud töös käsitletud nõuded on ainult algne versioon loodavast rakendusest. Tulevikus on soov rakenduse võimalusi laiendada. Esiteks on üks kliiniku teenustest töötubade läbiviimine. Enne pakutavaid töötubasid on vajalik luua küsimustikud, et saada parem

ülevaade osavõtjate taustast. Varasemalt on see toimunud nii, et saadetakse meili teel laiali failid ning spetsialist peab ise järke pidama, kes on küsimustikele vastanud ning milline on osalejate taust. Küsimustike loomise ja vastamise võimalus muudaks spetsialistide töö palju kiiremaks ning osalejatele vastamise mugavamaks.

Lisaks visiitide läbiviimisele on oluline, et patsiendid teeksid erinevaid harjutusi ka kodustes tingimustes, et teraapia tulemus oleks võimalikult efektiivne. Selleks, et patsientidel oleks hea ülevaade, milliseid harjutusi teha kodus, on hea tulevikus luua harjutuste pank, kus on olemas sõltuvalt probleemile vastav harjutus. Spetsialistidel oleks võimalik videod üles laadida ning patsiendil neid vaadata.

Projekti eduka käivituse korral on tulevikus võimalik antud rakendust kohandada ka mõne muu eriala spetsialistidele. Näiteks füsioterapeutidel võib sarnaselt olla vajadus, et patsientidele erinevaid harjutusi üles laadida ning paluda neil kodus tehtud harjutusi üles laadida [37].

Tehniliselt on oluline, et testide osakaal projektis suureneks. Seda eelkõige projekti töökindluse tagamiseks. Rakenduses võiks lisaks ühiktestidele olla ka läbivtestid. Viimaks võiks lisada integratsioonitestid.

7 Kokkuvõte

Töö alguses anti ülevaade lahendatavast probleemist ja selle aktuaalsusest ning uuele rakendusele esitatavatest ärinõuetest. Probleem oli olemasoleva rakenduse arhitektuur, oluliste võimaluste puudumine ning arenduse jätkusuutlikkus. Seejärel analüüsiti rakenduse tehnilisi nõudeid, millele järgnes praktilise lahenduse teostamine. Viimaks kirjeldati tulemusi.

Pärast ärinõuete väljaselgitamist analüüsiti, kuidas on antud nõudeid kõige otstarbekam tehniliselt teostada. Lisaks nõuetele tuli arvesse võtta seadusest tulenevaid aspekte ning kliinikus olemasolevaid võimalusi.

Rakenduse jätkusuutliku arenduse tagamiseks analüüsiti peatükis 4.5, mis tehnoloogiad on laialdaselt Eestis kasutusel ning soodustaksid tulevikus refaktoreerimist ja uute võimaluste lisamist. Lisaks sellele toodi peatükis 4.6 välja, kuidas aitab pidev integratsioon ja pidev evitamine kaasa rakenduse jätkusuutlikule arendusele.

Töö tulemuseks on uus rakendus, mis lähiajal asendab olemasoleva rakenduse. Uues äpis on täiendavad meetmed turvalisuse tagamiseks ning uute kasutajate registreerimine toimub automaatselt. Teiseks on loodud võimalus audio- ja videofaile veebilehitsejas vaadata ning patsiendil ja spetsialistil on võimalik nendele kommentaare lisada.

Lisaks olemasolevate võimaluste täiustamisele on uues rakenduses loodud mitu uut funktsionaalsust. Kõige olulisem neist on võimalus videosilla teel kaugteraapia läbiviimine. Veel saab välja tuua paranenud kasutajamugavuse, mis saavutati tänu uue rakendusliidese arendusele ning mitme keele toele.

Kasutatud kirjandus

- [1] World Economic Forum, „Shift to digital during the pandemic could enable universal health coverage,“ [Võrgumaterjal]. Kättesaadav: <https://www.weforum.org/agenda/2020/12/shift-to-digital-during-pandemic-could-enable-universal-health-coverage/>. [Kasutatud 19.04.2021].
- [2] Eesti Haigekassa, „Kaugteenuste arendamine,“ [Võrgumaterjal]. Kättesaadav: <https://www.haigekassa.ee/en/node/163842>. [Kasutatud 18.04.2021].
- [3] E. T. T. P. Susan Howell, „Delivering the Lee Silverman Voice Treatment (LSVT) by web camera: a feasibility study,“ *International Journal of Language and Communication Disorders*, kd. 44, nr 3, pp. 287-300, 2009.
- [4] World Health Organization, „Digital health,“ [Võrgumaterjal]. Kättesaadav: <https://www.who.int/health-topics/digital-health>. [Kasutatud 18.04.2021].
- [5] J. Ingeno, *Software Architect Handbook*, Packt Publishing, 2018.
- [6] OWASP, „OWASP Top Ten,“ [Võrgumaterjal]. Kättesaadav: <https://owasp.org/www-project-top-ten/>. [Kasutatud 19.04.2021].
- [7] Riigi Teataja, „Isikuandmete kaitse seadus,“ [Võrgumaterjal]. Kättesaadav: <https://www.riigiteataja.ee/akt/104012019011>. [Kasutatud 28.04.2021].
- [8] M. Julian, *Practical Monitoring*, O'Reilly Media, Inc., 2017.
- [9] Riigi Teataja, „Tervishoiuteenuste korraldamise seadus,“ [Võrgumaterjal]. Kättesaadav: <https://www.riigiteataja.ee/akt/121042021016>. [Kasutatud 21.04.2021].
- [10] B. G. D. S. D. W. K. J. Raj Bala, „Gartner,“ 1 September 2020. [Võrgumaterjal]. Kättesaadav: <https://www.gartner.com/doc/reprints?id=1-1ZDZDMTF&ct=200703&st=sb>. [Kasutatud 28.03.2021].
- [11] WHATWG, „HTML Living Standard,“ [Võrgumaterjal]. Kättesaadav: <https://html.spec.whatwg.org/>. [Kasutatud 17.04.2021].
- [12] Adobe Inc., „Adobe Flash Player EOL,“ [Võrgumaterjal]. Kättesaadav: <https://www.adobe.com/products/flashplayer/end-of-life.html>. [Kasutatud 05.05.2021].
- [13] Google Developers, „WebTRC,“ [Võrgumaterjal]. Kättesaadav: <https://webtrc.org/>. [Kasutatud 13.05.2021].
- [14] Spring, „Spring Framework Overview,“ [Võrgumaterjal]. Kättesaadav: <https://docs.spring.io/spring-framework/docs/current/reference/html/overview.html#spring-introduction>. [Kasutatud 28.03.2021].
- [15] SK ID Solutions, „Mobile-ID Java client,“ [Võrgumaterjal]. Kättesaadav: <https://github.com/SK-EID/mid-rest-java-client>. [Kasutatud 05.05.2021].
- [16] Google Developers, „Progressive Web Apps,“ [Võrgumaterjal]. Kättesaadav: <https://web.dev/progressive-web-apps/>. [Kasutatud 17.04.2021].

- [17] Stack Overflow, „Developer Survey 2020,“ 2020. [Võrgumaterjal]. Kättesaadav: <https://insights.stackoverflow.com/survey/2020#technology-programming-scripting-and-markup-languages-professional-developers>. [Kasutatud 17.04.2021].
- [18] J. O. C. Jeff Sutherland, A Scrum Book, Pragmatic Bookshelf, 2019.
- [19] K. M. Sean P. Kane, Docker: Up & Running, 2nd Edition, O'Reilly Media, Inc, 2018.
- [20] M. Churchman, „DevOps and Containers, On-Prem or in the Cloud,“ 26.01.2021. [Võrgumaterjal]. Kättesaadav: <https://rancher.com/devops-containers-prem-cloud>. [Kasutatud 20.04.2021].
- [21] GitHub, Inc., „Pushing and pulling Docker images,“ [Võrgumaterjal]. Kättesaadav: <https://docs.github.com/en/packages/guides/pushing-and-pulling-docker-images>. [Kasutatud 26.04.2021].
- [22] D. Sato, „Canary Release,“ [Võrgumaterjal]. Kättesaadav: <https://martinfowler.com/bliki/CanaryRelease.html>. [Kasutatud 26.04.2021].
- [23] M. Amundsen, Design and Build Great Web APIs, Pragmatic Bookshelf, 2020.
- [24] React, „Context,“ [Võrgumaterjal]. Kättesaadav: <https://reactjs.org/docs/context.html>. [Kasutatud 12.05.2021].
- [25] Spring, „Profiles,“ [Võrgumaterjal]. Kättesaadav: <https://docs.spring.io/spring-boot/docs/current/reference/html/spring-boot-features.html#boot-features-profiles>. [Kasutatud 01.05.2021].
- [26] E. Gamma, R. Johnson, R. Helm ja J. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley Professional, 1994.
- [27] Mozilla, „Common MIME types,“ [Võrgumaterjal]. Kättesaadav: https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/MIME_types/Common_types. [Kasutatud 25.04.2021].
- [28] XTech, „Upload API,“ [Võrgumaterjal]. Kättesaadav: <https://ant.design/components/upload/#API>. [Kasutatud 25.04.2021].
- [29] Video.js, „Audio Tracks,“ [Võrgumaterjal]. Kättesaadav: <https://docs.videojs.com/tutorial-audio-tracks.html>. [Kasutatud 23.04.2021].
- [30] Video.js, „Skins,“ [Võrgumaterjal]. Kättesaadav: <https://docs.videojs.com/tutorial-skins.html>. [Kasutatud 23.04.2021].
- [31] Amazon, „Sharing an object with a presigned URL,“ [Võrgumaterjal]. Kättesaadav: <https://docs.aws.amazon.com/AmazonS3/latest/userguide/ShareObjectPreSignedURL.html>. [Kasutatud 23.04.2021].
- [32] Oracle, „Core J2EE Patterns - Data Access Object,“ [Võrgumaterjal]. Kättesaadav: <https://www.oracle.com/java/technologies/dataaccessobject.html>. [Kasutatud 25.04.2021].
- [33] Twilio Inc., „Default Room Settings,“ [Võrgumaterjal]. Kättesaadav: <https://www.twilio.com/docs/video/javascript-getting-started#default-room-settings>. [Kasutatud 11.05.2021].
- [34] Twilio Inc., „Rest API: Recordings,“ [Võrgumaterjal]. Kättesaadav: <https://www.twilio.com/docs/video/api/recordings-resource>. [Kasutatud 11.05.2021].

- [35] React-i18next, „Multiple Translation Files,“ [Võrgumaterjal]. Kättesaadav: <https://react.i18next.com/guides/multiple-translation-files>. [Kasutatud 23.04.2021].
- [36] XTech, „Internationalization,“ [Võrgumaterjal]. Kättesaadav: <https://ant.design/docs/react/i18n>. [Kasutatud 27.04.2021].
- [37] edu.ee, „Füsioterapeut,“ [Võrgumaterjal]. Kättesaadav: <https://haridusportaal.edu.ee/ametialad/f%C3%BCsioterapeut#t%C3%B6%C3%B6%C3%BClesanded>. [Kasutatud 02.05.2021].

Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks¹

Mina, Tristan Krass

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Rakendus kõnehäirete diagnostika ja kaugteraapia läbiviimise toetamiseks“, mille juhendaja on Märt Kalmo, MSc.
 - 1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
 - 1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

15.05.2021

¹ Lihtlitsents ei kehti juurdepääsupiirangu kehtivuse ajal vastavalt üliõpilase taotlusele lõputööle juurdepääsupiirangu kehtestamiseks, mis on allkirjastatud teaduskonna dekaani poolt, välja arvatud ülikooli õigus lõputööd reprodutseerida üksnes säilitamise eesmärgil. Kui lõputöö on loonud kaks või enam isikut oma ühise loomingu tegevusega ning lõputöö kaas- või ühisautor(id) ei ole andnud lõputööd kaitsvale üliõpilasele kindlaksmääratud tähtajaks nõusolekut lõputöö reprodutseerimiseks ja avalikustamiseks vastavalt lihtlitsentsi punktidele 1.1. ja 1.2, siis lihtlitsents nimetatud tähtaja jooksul ei kehti.

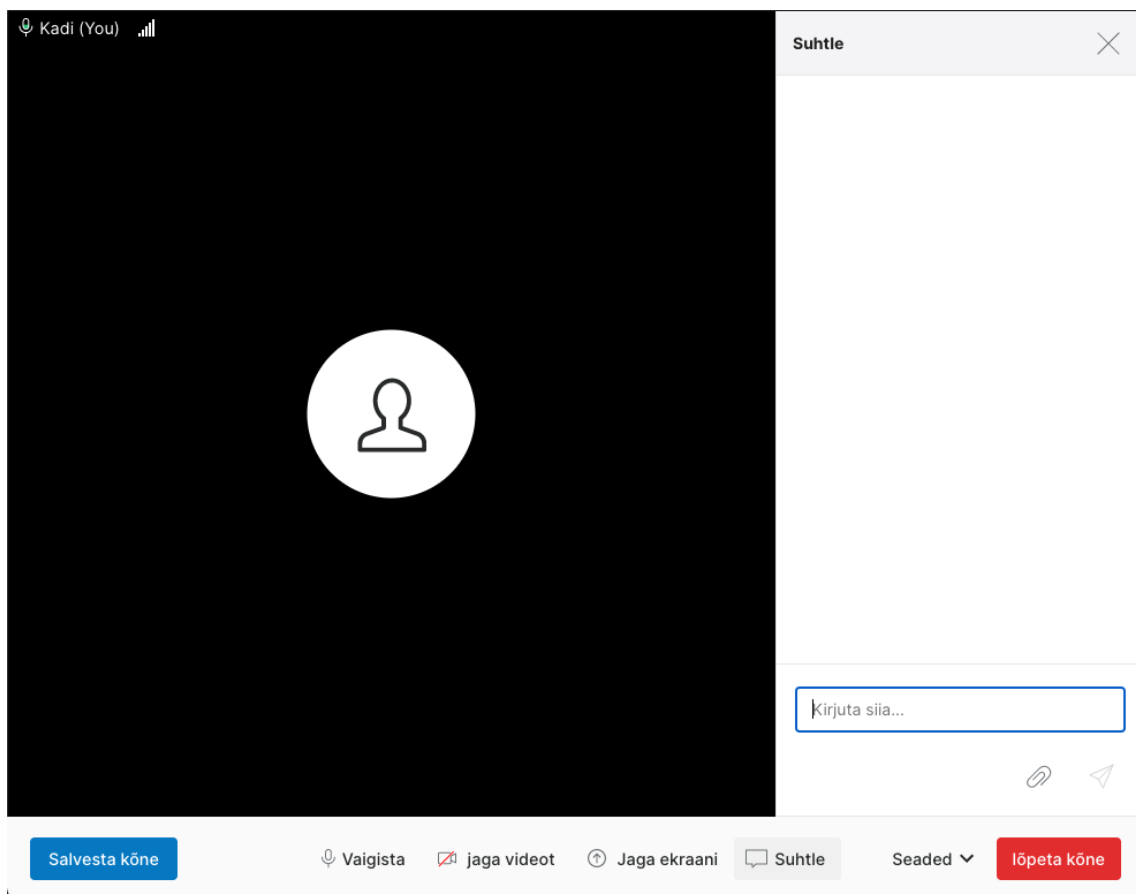
Lisa 2 Spetsialistide vaade failidest

Kõik patsiendid Sinu patsiendid

Eesnimi	Perenimi	Vanus	Märgendid	Fail
Mavis	Borer	34	epikriis r-häälik	Vaata videot
Amelia	Reichert	24	4-aastane 4-aastane	Vaata videot
Lia	Kautzer	23	kõne r-häälik häälik	Vaata videot
Daryl	Oberbrunner	21	epikriis kõne	Vaata videot
Justyn	Daugherty	13	5-aastane epikriis	Vaata videot
Samanta	Hamill	36	kõne 5-aastane 4-aastane	Vaata videot
Dangelo	Swift	32	epikriis 5-aastane	Vaata videot
Susanna	Tremblay	21	kõne epikriis	Vaata videot
Janice	Dach	10	4-aastane r-häälik kõne	Vaata videot
Adolfo	Larson	48	epikriis 4-aastane	Vaata videot

< 1 >

Lisa 3 Videokõne vaade rakenduses



Lisa 4 Patsiendi tervikprofili nägemise võimalus

[Info patsiendi kohta](#) Kaugteraapia sessioonid

Eesnimi	Perenimi	E-post	Vanus	Aadress	Teraapiate arv
Mari	Kalamari	marikalamari@hotmail.com	32	Vana-Kalamaja 12b	5

▼ Ajajoon

- Liitus lehega 05.06.2015
- Esimene visiit Kadi Mari juures 20.05.2015
- Teine visiit Kadi Mari juures 20.06.2015
- Kolmas visiit Kadi Mari juures 20.07.2015
- Neljas visiit Kadi Mari juures 20.08.2015
- Viies visiit Kadi Mari juures 20.09.2015

▼ Tegevused

[Saada sõnum](#)