

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Eliise Tall 205896IABB

Veebirakendus enesetapukriisi ennetamiseks

Bakalaureusetöö

Juhendaja: Meelis Antoi
Magistrikraad

Tallinn 2024

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: Eliise Tall

14.05.2024

Annotatsioon

Eesti enesetapunäitajad on teiste Euroopa riikidega võrreldes väga kõrged ja samal ajal on puudus abi pakkumiseks sobivatest eestikeelsetest vahenditest. Turvaplaan on levinud ja tõhus töövahend, mis toetab enesejuhtimist ning pakub tuge inimestele, kes kogevad enesetapumõtteid. Bakalaureusetöö eesmärgiks oli luua veebirakendus, mis võimaldaks kliendi ja spetsialisti koosarendatava turvaplaani loomist.

Eesmärgi saavutamiseks viidi läbi küsitlus tellijaga, analüüsiti temaatilist kirjandust, probleemi ja alternatiivseid lahendusi. Nende tegevuste põhjal selgitati välja, milline võiks olla uus lahendus probleemile. Seejärel valiti parimad tööriistad ja vastavalt nõuetele loodi rakendus, mis läbis testimise ja mille tulemusi analüüsiti. Enesetapumõtetega klientide puhul on kõige olulisem abi kiire ja lihtne kättesaadavus. Rakendus tagab vaimse tervise spetsialistile ja kliendile juurdepääsu enesetapukriisi turvaplaanile, olles kättesaadav igal ajal ja igas kohas.

Bakalaureusetöö eesmärk sai täidetud - loodud enesetapukriisi turvaplaan võimaldab lihtsalt ja mugavalt jälgida kliendi progressi ja plaani kasutamissagedust ning seeläbi vajadusel operatiivselt sekkuda enesetapukriisi süvenemisesse. Rakendus võimaldab kliente individuaalselt toetada ja juhendada nende enesetapumõtete või -käitumise korral.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 25 leheküljel, 7 peatükki, 7 joonist.

Abstract

Online application for the prevention of suicidal crisis

Suicide rate in Estonia is one of the highest compared to other European countries. At the same time, there is a lack of suitable tools in Estonian to provide support for the suicidal. Suicide safety plan is common and effective tool that supports self-management and provides guided help for people experiencing suicidal thoughts. The aim of the bachelor's thesis was to create a web application that would enable the creation of suicide safety plan developed together by the client and the specialist.

To achieve the goal, a customer survey was conducted, previous research, literature, the problem and alternative solutions were analyzed. Based on these activities, it was established, what a new solution to the problem could be. Then the best tools were selected and an application was created according to requirements, which was tested and the results of which were analyzed. The most important action for clients with suicidal thoughts is quick and easy access to help. The created app provides mental health professionals and clients access to a suicide crisis safety plan, accessible anytime, anywhere.

The aim of the bachelor's thesis was fulfilled – the developed safety plan for suicide crisis allows easy and convenient monitoring of the client's progress and frequency of use of the plan, and thus, if necessary, operatively intervene in the worsening of the suicide crisis. The application allows clients to be individually supported and guided in case of suicidal thoughts or behavior.

The thesis is in Estonian language and contains 25 pages of text, 7 chapters, 7 figures.

Lühendite ja mõistete sõnastik

API	Application Programming Interface
CSS	Cascading Style Sheets
DRY	Don't repeat yourself
HTML	HyperText Markup Language
HTTP	Representational State Transfer
JSON	JavaScript Object Notation
ORM	Object-relational Mapper
REST	Representational State Transfer
RoR	Ruby on Rails
SOAP	Simple Object Access Protocol
SQL	Structured query language
SPA	Single-page application
URL	Uniform Resource Locator
XML	Extensible Markup Language

Sisukord

1 Sissejuhatus	9
1.1 Probleem	9
1.2 Eesmärk	10
1.3 Metoodika.....	10
2 Eksisteerivad lahendused.....	11
2.1 Safety Plan, Safety Net ja Be Safe	11
2.2 Stay Alive	11
2.3 Staying Safe ja My Safety Plan	12
2.4 IntroScope.....	12
2.5 Üksildust leevendav eneseabirakendus	12
3 Analüüs.....	13
3.1 Nõuete määramine	13
3.1.1 Funktsionaalsed nõuded	13
3.1.2 Mittefunktsionaalsed nõuded.....	14
3.2 Veebirakenduse arhitektuur	17
3.3 Serveripoolne tehnoloogia.....	18
3.4 Kliendipoolne tehnoloogia	19
3.5 Versioonihaldus	20
3.6 Andmebaas	20
3.7 Docker	21
3.8 Objekti detailne kirjeldus.....	21
4 Tulemuste analüüs	22
4.1 Üldine arhitektuur.....	22
4.2 Klientrakenduse arhitektuur	23
4.3 Serverirakenduse arhitektuur	24
4.4 Veebirakenduse disain.....	25
4.5 Ärijuhtumi sisend	27
5 Järeldused	30

6 Võimalused edasiarenduseks	32
7 Kokkuvõte	33
Kasutatud kirjandus	34
Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks	37

Jooniste loetelu

Joonis 1. Turvaplaani skeem	15
Joonis 2. Infokaardi skeem	15
Joonis 3. Jagatud plaanide skeem	16
Joonis 4. Lingid kliendi riskitulemuste, infokaardi ja turvaplaani lehele	16
Joonis 5. Vea sõnum plaani jagamisel	25
Joonis 6. Jagatud plaanide vaade	26
Joonis 7. Turvaplaani vaade nutitelefonist	27

1 Sissejuhatus

Eesti enesetapunäitajad on teiste Euroopa riikidega võrreldes väga kõrged [1]. Suitsiidimõtetega inimese lähedased on sellest olukorrast samuti mõjutatud [2]. Uuringute järgi on iga suitsiid riivanud keskmiselt 135 enesetapu sooritanu lähedast [2]. Suitsiidsete mõtete tekkimiseks võib olla erinevaid põhjuseid. Mõneks riskiteguriks on näiteks sotsiaalne tõrjutus, krooniline või parandamatu kehaline haigus, halvenenud suhted vanematega, lahkuminek, rahalised raskused, töö kaotus, kiusamine ja palju muud. Suurema osa enesetapu sooritajaid on varasemalt oma elus saanud psüühikahäire diagnoosi [3]. Kõige sagedasem diagnoos on depressioon, kuid arvestatav osakaal enesetapujuhtumitest on seotud ka alkoholi ning skisofreeniaga [3]. Eestis kogeb aasta jooksul vaimse tervise probleeme iga neljas inimene ja igal aastal jääb kuni 200 000 inimest ilma vaimse tervise toest [4]. Suured enesetapunäitajad viitavadki sellele, et inimene on jäänud ilma õigeaegsest ja vajalikust abist, sest valdav osa suitsiide on siiski ennetatavad [5]. Abi pakkumiseks jääb sobivatest – lihtsalt ja kiirelt ligipääsetavatest - vahenditest puudu.

1.1 Probleem

Enesetapukriisi turvaplaan on levinud ja tõhus töövahend, mis toetab enesejuhtimist ning pakub tuge inimestele, kes kogevad enesetapumõtteid. Turvaplaan on struktureeritud ning sammhaaval läbimiseks. Plaan käsitleb kriisi alguse ja süvenemise ohumärke, isiklikke toimetulekustrateegiaid, mõtete kõrvale juhtimist, lähedaste kontakte, tervishoiuspetsialistide kontakte ja ümbruse turvalisuse suurendamise võimalusi. Enesetapumõtetega inimene saab koos oma nõustaja, sõbra või muu usalduslikuga läbi arutada, millised valikud on tema jaoks sobivad ja plaani koostada [6].

Seni eksisteerivate lahenduste suurimaks puudujäägiks on mugava koostöövõimaluse puudumine. Ei saa jagada plaani viisil, et ka teised, näiteks nõustajad saaksid üksikut plaani muuta ja selle loomise kulgu lihtsasti jälgida. Praegu eksisteerivad lahendused, kus turvaplaani saab muuta ainult üks inimene. Samuti on puuduseks kehv kättesaadavus.

Oma plaanile ei saa erinevatest brauseritest ja seadmetest ligi viisil, et seda ka edasi saaks arendada. Plaani muutmine on aga oluline, sest inimese elu on muutumises ja ka turvaplaan peaks olema dünaamiline ning võimaldama täiendusi ja muudatusi lisada. Üheks suureks miinuseks võib märkida veel asjaolu, et alternatiivsete lahenduste seas ei leidu ühtegi, mis oleks suunatud eesti keelt kõnelevale ja Eestis asuvale inimesele.

1.2 Eesmärk

Bakalaureusetöö eesmärgiks on luua veebirakendus, mis võimaldaks koosarendatava enesetapakriisi turvaplaani loomist. Veebirakenduses saab luua kasutajakonto ning koostada turvaplaan, mida nõustajad saavad teha koos infokaardi ja riskihindamise tulemustega. Turvaplaani on võimalik teiste kasutajatega jagada ja ühiselt arendada. Uus lahendus on suunatud igähele, kes soovib turvaplaani koostada, kuid oluliseks sihtrühmaks on Sotsiaalkindlustusameti ohvriabi kriisitelefoni nõustajad, kes oma töö iseloomu tõttu suitsiidsete inimestega igapäevaselt kokku puutuvad ning sellist veebirakendust oma igapäevases töös vajaksid ja kasutaksid.

1.3 Metoodika

Eesmärgi saavutamiseks tuleb esmalt viia läbi küsitlus tellijaga, uurida selle teemalist kirjandust, probleemi ja alternatiivseid lahendusi, mis on kirjeldatud töö sissejuhatuses. Nende tegevuste põhjal selgitatakse, milline võiks olla uus lahendus probleemile. Seejärel tehakse parimate tööriistade valik, mida sisaldab töö metoodika osa. Vastavalt nõuetele luuakse rakendus, mida testitakse ja mille tulemusi analüüsitakse.

2 Eksisteerivad lahendused

Turvaplaani koostamiseks on loodud erinevaid vahendeid. Järgmisena analüüsitakse saadaolevaid alternatiive uuele lahendusele ning varasemaid sama valdkonna bakalaureusetöid. Alternatiivideks on valitud vahendid, mis on võimalikud sarnased käesoleva bakalaureusetöö uue lahendusega.

2.1 Safety Plan, Safety Net ja Be Safe

Safety Plan, Safety Net ja Be Safe on lahenduse poolest väga sarnased ingliskeelsed mobiilirakendused, mis on loodud selleks, et kasutaja saaks turvaplaani etappide kaupa läbida. Rakenduse installinud kasutajal on võimalus plaanis pidevalt erinevaid muudatusi teha. Konto loomine pole nende mobiilirakenduste kasutamiseks ette nähtud, seetõttu puudub võimalus teises seadmes või brauseris oma plaani edasi koostada, muuta ja vaadata. Kasutaja saab läbi mobiilirakenduste telefoni teel nõustaja või enda sisestatud kontaktiga ühendust võtta. Safety Plan ja Be Safe rakenduses on see tegevus võimalik ka meili teel.

Suicide Safety Plan rakenduses saab lisaks avada ka Google Mapsi (kaarditarkvara) kliinikumi ja keskuste asukohtadega. Samuti saab plaani jagada eelistatud suhtlustarkvara kaudu, näiteks Facebookis, Messengeris, Teamsis ja WhatsAppis, mida Be Safe ja Safety Net mobiilirakenduses teha ei saa.

2.2 Stay Alive

Stay Alive on konfidentsiaalne mobiilirakendus, mis on arendatud Inglismaa heategevuse poolt ning suunatud inimestele kelle lähedane või kes ise kogevad suitsiidseid mõtteid. Rakendus on kõigile tasuta kättesaadav ning reklaamivaba. See on saadaval 14 keeles, seal hulgas bulgaaria, taani, inglise, soome, prantsuse, saksa, norra, rumeenia, vene, hispaania, rootsi, poola ja kõmri keel. Selles rakenduses on võimalik luua konto ning kasutaja saab seeläbi ka teisest seadmest oma koostatud plaanile ja sisule ligi. Peale

turvaplaani saab seal veel inspireerivaid pilte lisada, heaolu plaani koostada ning hingamisharjutusi järgida. Samuti on rakenduses mitmeid nõuandematerjale ja juhendeid. Peale telefoni ja emaili on loodud otseteed ka erinevate keskuste veebisaitidele. Plaani saab jagada erinevate suhtlustarkvarade kaudu PDF kujul.

2.3 Staying Safe ja My Safety Plan

Staying Safe ja My Safety Plan on inglisekeelsed veebilehed turvaplaani koostamiseks. Muudatusi juba koostatud plaanis saab teha ainult ühes seadmes ja brauseris. Kui plaan on koostatud saab selle alla laadida PDF kujul. Veebilehel on loodud ka otseteed, Staying Safe lehel UK piirkonna kiirkontaktidele ning My Safety Plan saidil Kanada piirkonna kontaktidele.

2.4 IntroScope

IntroScope on Kadri Jõe bakalaureusetöö [7] raames valminud eneseanalüüsi veebirakenduse prototüüp. Turvaplaani seal koostada ei saa, küll aga on seal võimalik mitmeid muid tegevusi teha, mida nõustajad vaimsete probleemide korral üldjuhul soovivad. Näiteks päeviku kirjutamine, une aja, tujude ja mõtete logimine ning unistuste ja eesmärkide seadmine. Loodud sisu saab rakendusesiseselt nõustajaga jagada, kuid koos seda muuta ei saa. Siiski on see nõustajale samuti hea töövahend, sest rakenduses on lihtne oma kliente ja nende vorme hallata.

2.5 Üksildust leevendav eneseabirakendus

Üksildust leevendav eneseabirakendus on Karl Hanseni bakalaureusetöö [8] raames teostatud mobiilirakenduse prototüüp. Rakendus on eelkõige suunatud inimestele, kes tunnevad end üksildasena, seega turvaplaani seal koostada ei saa, kuid langeb sama kategooria alla ehk on enesejuhtimisele suunatud vahend. Selles loodud sisu rakendusesiseselt kellegi teisega, näiteks nõustajaga jagada ei saa ehk rakendus on mõeldud ainult personaalseks kasutamiseks. Kasutajal on võimalus erinevaid pilte, videoid ja häälsõnumeid omavahel kombineerida ja seadistada need ajapeale käivituma, mis kokkuvõttes peaks tekitama tunde lähedase kohalolekust.

3 Analüüs

Selles peatükis analüüsitakse töövahendite valikut veebirakenduse teostamiseks käesoleva bakalaureusetöö raames.

3.1 Nõuete määramine

Nõuete määramisel on arvesse võetud kõiki tegevusi, mida ohvriabitöötajad suitsidaalse pöörduja kõne saades läbi teevad. Ohvriabi on ööpäevaringselt nii eesti, vene kui ka inglise keeles nõustamist pakkuv avalik ja maksuvaba sotsiaalteenus Eestis, et säilitada või parandada ohvri toimetulekuvõimet [9]. Nõuete määramiseks vajalik informatsioon saadi kaugnõustamisteenuse juhiga tehtud küsitluse tulemusena.

3.1.1 Funktsionaalsed nõuded

Kliendi funktsionaalsed nõuded

- Kliendina saan konto registreerida.
- Kliendina saan loodud kontoga sisse ja välja logida.
- Kliendina saan muuta oma kontoga seotud isikuandmeid.
- Kliendina saan turvaplaani koostada.
- Kliendina saan turvaplaani nõustajaga jagada.
- Kliendina näen jagatud turvaplaanidega seotud kasutajaid.
- Kliendina saan infokaarti vaadata.
- Kliendina saan riskitulemusi vaadata.
- Kliendina saan riski hinnata.

Nõustaja funktsionaalsed nõuded

- Nõustajana saan konto registreerida.
- Nõustajana saan loodud kontoga sisse ja välja logida.
- Nõustajana saan muuta oma kontoga seotud isikuandmeid.
- Nõustajana saan turvaplaani koostada.
- Nõustajana saan koostada infokaarti.
- Nõustajana saan riski hinnata.
- Nõustajana saan kirja panna riskihindamise tulemusi.
- Nõustajana näen jagatud turvaplaanidega seotud isikute kontaktandmeid.
- Nõustajana saan turvaplaane korduvalt muuta ja täiendada.
- Nõustajana saan infokaarte korduvalt muuta ja täiendada.
- Nõustajana saan ülevaate kõikidest endaga seotud infokaartidest, turvaplaanidest ja tulemustest.

3.1.2 Mittefunktsionaalsed nõuded

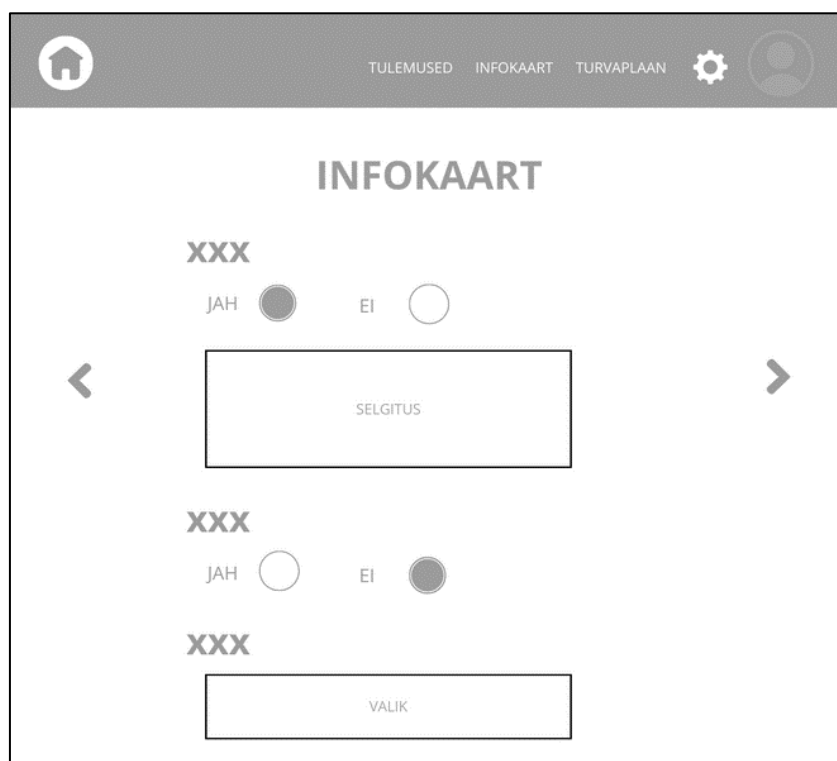
- Veebirakendus peab jälgima isikuandmete reegleid.
- Veebirakenduse sisu peab vastama eesti keele nõuetele.
- Veebirakenduse vaade peab kohanduma ekraani suurusega.

Peale nõuete väljaselgitamist loodi visuaalsed skeemid, mis aitavad määratleda lehekülgede kujunduse ja sisu.

Joonisel 1 on näha turvaplaani skeemi nõustaja vaates. Nõustaja saab navigeerida oma jagatud plaanide lehele.



Joonis 1. Turvaplaani skeem



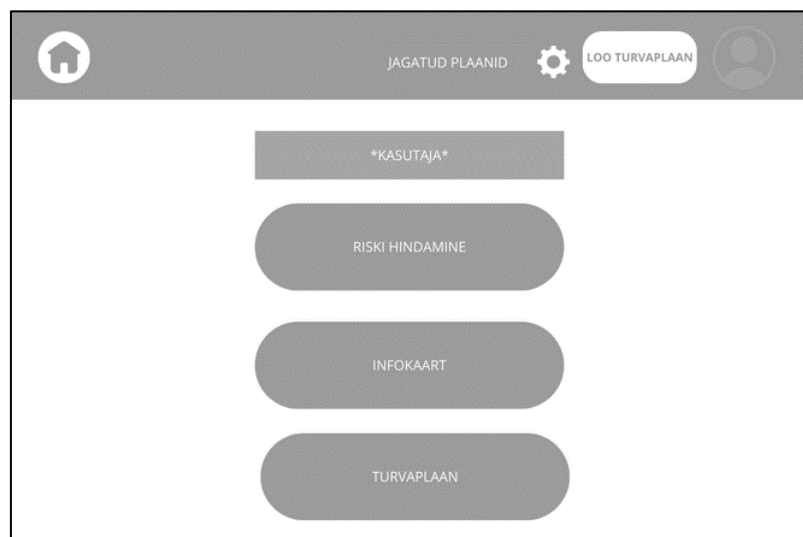
Joonis 2. Infokaardi skeem

Joonis 2 illustreerib infokaardi vaadet veebirakenduses, samuti on näha navigeerimisribal olevaid linke, mis on vajalikud kliendile, näiteks turvaplaan.



Joonis 3. Jagatud plaanide skeem

Joonis 3 illustreerib jagatud plaanide vaadet, mis on näha ainult nõustaja rollis olevale veebirakenduse kasutajale. Nõustaja saab ülevaate kõikidest klientidest, kes temaga oma turvaplaani jaganud on.



Joonis 4. Lingid kliendi riskitulemuste, infokaardi ja turvaplaani lehele

Joonisel 4 on näha vaadet, mida nõustajale kuvatakse, kui klõpsab ühele oma klientidest jagatud plaanide vaates.

3.2 Veebirakenduse arhitektuur

Veebirakenduste struktureerimiseks on kaks põhilist viisi. Üheks võimaluseks on käivitada kogu rakendus serveris. Teiseks variandiks oleks selle protsessi jagamine serveri ja kliendi (brauseri) vahel [10]. SPA ehk üheleherakenduse kontseptsioon on populaarne lähenemine veebirakendustele [11]. SPA rakendus laeb ainult teavet, mida kasutaja vajab [12]. Esialgsel käivitamisel saadetakse terve HTML leht ning hiljem uuendab server seda vastavalt kasutaja interaktsioonidele rakendusega [12]. JSON-failide kujul edastatakse vaid vajaminevad andmed, seega tervikuna lehte enam uuesti laeta [12].

SPA rakenduse kaks põhilist komponenti on klient ja server [13]. Klient on veebirakenduse visuaalne ja interaktiivne pool, mis sisaldab kõike, mida kasutaja vahetult kogeb, näiteks interaktiivsed nupud ja navigeerimismenüü [14]. Server on veebirakenduse osa, mis vastutab andmete salvestamise eest andmebaasi ning selle eest, et kõik kliendi poolel ootuspäraselt toimiks [14]. Server suhtleb kliendiga, edastab ja võtab vastu teavet, mida brauseris kuvatakse [14]. Veebirakenduse funktsionaalsus sõltub palju sellest, kui tõhusalt klient ja server omavahel suhtlevad ning ühe üksusena toimivad [14]. Selleks, et nende vaheline suhtlus toimiks, arvestavad nad mingi kindlaksmääratud suhtlusskeemi ehk protokolliga, mis määrab, kuidas andmeid edastatakse [15].

Veebirakenduses määrab kliendi ja serveri suhtluse HTTP-nimeline protokoll [15]. REST arhitektuuril põhinev API kasutab samuti HTTP protokoll, mis on oluline komponent paljudes APIdes [16]. REST on tänapäeval laialdaselt kasutatav arhitektuuristiil. Selle kõige olulisem eelis on paljude keelte ja platvormide koostalitlusvõime [10]. Võrreldes teiste HTTP protokolliga kasutatavate APIdega nagu SOAP, mis kasutab andmevahetuseks XMLi, on REST API lihtsam, standardiseeritud ning parema jõudlusega ka komplekssemate ja keerulisemate rakenduste puhul [17]. Samas, teatud olukordades, kus on vaja rangeid turvanõudeid võib SOAP või mõni muu lähenemisviis olla sobivam. Näiteks juhul, kui rakenduses on väga palju erinevaid autoriseerimismeetodeid [16].

Tüüpilist veebirakendust iseloomustab kolmekihiline arhitektuur, mis hõlmab esitusloogika kihti kliendi poolel, äri loogika kihti serveri poolel ja andmebaasi [13].

3.3 Serveripoolne tehnoloogia

Serverirakenduse arendamiseks on mitmeid programmeerimiskeeli, millest populaarseimad on Python, Ruby, PHP, C# [11]. Valmis lahenduse saavutamiseks on vaja ka raamistikku, mis on igal eelnevalt mainitud keelel erinev.

C# programmeerimiskeeles veebirakenduse loomiseks on ASP .NET Core raamistik, mis võimaldab luua HTTP põhiseid APIsid [18]. C# programmeerimiskeel on alati toetanud objektorienteeritud programmeerimist, samuti peaks sellist paradigmat rakendama ka iga .NET (Core) raamistikku kasutatav rakendus. Objektorienteeritud programmeerimine on oluline, kuna kood läheneb reaalelulistele kontseptsioonidele, muutes selle lihtsamini mõistetavaks ja tõlgendatavaks [10].

Ruby on Rails raamistik on peamiseks valikuks Ruby programmeerimiskeeles veebirakenduste arendamiseks. Ruby on samuti objektorienteeritud keel. Peale selle peetakse seda väga arendajasõbralikuks keeleks [19] ning RoR kasutab hästi ära selle programmeerimiskeele potentsiaali. Arendajad eelistavad RoRi eelkõige sisseehitatud testimisfunktsiooni tõttu, mis võimaldab kirjutada näiteks ühikteste suurema tõhususega, tagades samal ajal, et rakenduse kvaliteet ja stabiilsus ei langeks [20].

Pythonis programmeerimiseks on populaarseimateks variantideks Django ja Flask. Django pakub täielikku tuge erinevatele andmebaasidele, mis teeb arendajatele ülemineku ühelt andmebaasilt teisele lihtsamaks [20]. Samuti pakub palju sisseehitatud turvafunktsioone, mis muudavad selle eelistatud valikuks turvaliste veebirakenduste arendamiseks [20]. Flask võimaldab arendajatel Pythonis HTTP päringuid ja vastuseid käsitleda. See pakub kasulikke tööriistade ja funktsioonide komplekti töökindlate, turvaliste ja reageerivate veebirakenduste loomiseks [20].

Laravel on üks populaarsemaid raamistikke veebirakenduste arendamiseks. See veebiraamistik toetab süntaksit, mis võimaldab arendajatel kirjutada puhast ja loetavat koodi [20]. Lai valik sisseehitatud funktsioone ja väljendusrikas süntaks toetavad marsruutimise, andmebaasipäringute ja autentimise sujuvamaks muutmisel, keskendudes rohkem lihtsusele ja loetavusele [20].

Lõputöö autoril on kõige rohkem kokkupuudet C# programmeerimiskeeles arendamisega ja tunneb seetõttu kõige paremini .NET Core raamistikku. Sellest lähtuvalt kasutatakse serverirakenduse teostamiseks just neid tehnoloogiaid.

3.4 Kliendipoolne tehnoloogia

Veebirakenduse arendamiseks on vaja praktilist raamistikku. Kliendipoolsed raamistikud koosnevad sageli kolmest põhitehnoloogiast, mida nimetatakse HTML-iks, CSS-iks ja JavaScriptiks. HTML-i kasutatakse teksti paigutamiseks lehele. CSS-i kasutatakse teksti stiliseerimiseks ning JavaScripti kasutatakse interaktiivsete efektide lisamiseks, näiteks jagamisnupp [21]. Nimetatud tehnoloogiaid kasutades saab luua veebirakendusi, mis töötavad erinevates veebibrauserites ja seadmetes ehk arendada platvormiüleseid rakendusi [22].

TypeScript on avatud lähtekoodiga programmeerimiskeel, mille on välja töötanud Microsoft, et kõrvaldada JavaScripti puudused. TypeScript on arendatud JavaScripti pealt ning see kompileerub puhtaks JavaScripti koodiks. TypeScript laiendab JavaScripti keelt, lisades sellele staatilise tüübikontrolli ja teisi omadusi, et vähendada keerukust ja parandada koodi kvaliteeti, mis on kasuks eriti suurtes projektides [22].

Kolm kõige populaarsemat varianti komplekssete, interaktiivsete ja kaasaegsete klientrakenduste loomiseks on JavaScripti teek React ja raamistikud Angular ning Vue.js. Need on komponendipõhised, TypeScripti toega ja võimaldavad kiiresti kasutajaliidese funktsioone luua. Siiski on neil erinev struktuur ja arhitektuur ning oma eelised. Angular on vastupidav ja olnud kasutusel kõige kauem, mistõttu tõestanud aja jooksul usaldusväärset ja efektiivset reaalsetes projektides. React on paindlik ja kiire ning Vue lihtne ja suure jõudlusega. Samuti on Vue progressiivne, lihtne integreerida ja sobib hästi üheleherakenduste arendamiseks [12].

Lõputöö autoril puudub kogemus React ja Angular raamistikega, mistõttu kasutatakse klientrakenduse arendamiseks Vue raamistikku, et arendusprotsess oleks piiratud aja tõttu võimalikult sujuv.

3.5 Versioonihaldus

Versioonihaldus on arenduse lahutamatu osa. Versioonihalduse süsteem aitab jälgida koodi edenemist arenduse jooksul, tagab juurdepääsu projekti erinevatele versioonidele ning projekti andmete salvestamise. Vea sattumisel koodi, saab naasta hilisemale versioonile koodist, samuti on võimalik teatud muudatusi tagasi võtta ilma, et kaoks kogu vahepealne töö. Projekti mis tahes osa puhul saab määrata, millal, miks ja kes seda kunagi muutis. Versioonihaldus võimaldab ühel inimesel kogu projekti uuesti kloonida ning selle arendamiseks mitut arvutit kasutada, seega on see kasulik ka siis, kui tiim puudub nagu ka käesoleva töö puhul [23].

Versioonihaldus hõlmab repositooriumit ehk hoidlat, mis on projekti muudatuste andmebaas, ning isiklikku koopiat kõigist projektis sisalduvatest failidest [23].

Kolm kõige tuntumat versioonihalduse tööriista on Git, Subversion ja Mercurial [23]. Git on neist kõige populaarsem valik. See on avatud lähtekoodiga süsteem, mida kasutatakse igas suuruses projektide jaoks, mis teeb sellest populaarse valiku nii idufirmade, ettevõtete ja kõige vahepealse seas [24].

3.6 Andmebaas

Peaaegu iga veebipõhine rakendus nõuab andmebaasi. Need saab laias laastus rühmitada relatsioonilisteks ja mitterelatsioonilisteks andmebaasideks [25]. Üks hoiab struktureeritud andmeid ja teine mitte, see on põhiline erinevus nende kahe tüübi vahel. Populaarseimateks andmebaasideks veebiarenduses on MySQL, Microsoft SQL Server, Oracle Database, MongoDB ja PostgreSQL, mis on tuntud kui ka Postgres [26]. Viimast kasutatakse käesoleva lõputöö raames valmivas veebirakenduses. Postgres on avatud lähtekoodiga relatsiooniline andmebaas veebirakenduste arendamiseks, mis on tuntud oma skaleeritavuse, töökindluse ja laiendatavuse poolest [25]. PostgreSQL on populaarsetel platvormidel nagu Windows, MacOS, Linux ja UNIX täiesti tasuta, ilma litsentsitasude või kasutuspiiranguteta [25]. PostgreSQLi tugevuseks on täiustatud SQLi tugi, mis aitab arendajatel töötada keeruka andmetöötuse ja -analüüsiga [25].

Lõputöö autoril on varasem kokkupuude postgres andmebaasiga ja seetõttu on see ka käesoleva töö veebirakenduse andmebaasiks.

3.7 Docker

Docker on rakenduse loomise, juurutamise ja käivitamise lihtsustamiseks kasutades selleks konteinereid. Konteiner võimaldab käivitada rakendusi virtuaalses keskkonnas, pakendades kokku kõik vajalikud elemendid, nagu failid, teegid ja muud olulised komponendid [27].

Võrreldes postgresi kohaliku installimisega on konteineri kasutamine lihtsam, lisakonfiguratsiooni tegemine ja oma taustaprotsesside käivitamine võtavad lisaega, nõuavad sügavamaid tehnilisi teadmisi ja ei kohane hästi muutuvate nõuetega [28].

3.8 Objekti detailne kirjeldus

Bakalaureusetöö raames arendatud veebirakendus on üheleherakendus, mis hõlmab endas klientrakendust, serverirakendust ning andmebaasi. Klientrakenduse realiseerimiseks valiti Typescriptiga Vue raamistik ja serverirakenduse realiseerimiseks .NET raamistik ning C# programmeerimiskeel. Andmete salvestamiseks on PostgreSQL andmebaas. Mainitud tehnoloogiad valiti vastavalt projekti vajadustele ning bakalaureusetöö autori teadmistele ja kompetentsile.

4 Tulemuste analüüs

Selles peatükis kirjeldatakse, milline on tehtud töö tulemus ja järeldused selle põhjal.

4.1 Üldine arhitektuur

Veebiprogrammeerimisel on levinud põhimõte, et rakenduse juurutamine tuleb jagada kolmeks [29]. Lõputöö raames valminud veebirakendus koosneb kolmest eraldiseisvast komponendist:

- **Klientrakendus** – vastutab kõige selle eest, mis on seotud rakenduse välimusega. Klientrakenduse ülesandeks on kuvada serverilt saadud andmeid atraktiivsel ja informatiivsel viisil. Samuti reageerida kasutaja toimingutele, milleks on näiteks teksti sisestamine ja nupule klõpsamine, et vastavalt sellele serverirakendusele päringuid saata.
- **Serverirakendus** – sisaldab endas kogu äriloogikat, mis määrab, kuidas peaks läbi klientrakenduse tulnud kasutajate päringutele reageerima, et veebirakenduse funktsionaalsus saaks ootuspärasel viisil toimida. Serverirakendus vastutab eelkõige andmetöötuse eest, andmete tagastamise eest klientrakendusele, samuti suhtleb andmebaasiga, kus andmeid hoitakse.
- **Andmebaas** – hoiab andmeid, mida sinna salvestatakse ja kasutaja poolt päritakse.

Kliendi ja serveri vaheline suhtlus toimub HTTP protokolliga kaudu. Klientrakenduselt saadud päringu osadeks on URL, mis määrab serveri ressursi, millele soovitakse ligi pääseda. Päringu osaks on veel HTTP verb GET, POST, PUT või DELETE, mis määrab nõutava toimingu. Lisaks võib päring sisaldada lisainformatsiooni, näiteks nimi-väärtus paaride kujul, mis on kodeeritud päringu URL-i otsa. Server töötleb

päringut ja vastab kliendile HTTP vastusega, mille osadeks on staatuskood ning GET meetodi puhul ka vastuse keha päritud ressursiga [30].

Staatuskood annab kliendile teada, mis serveri poolt töödeldud päringuga juhtus. Valminud veebirakenduses võib esineda näiteks staatuskoodi 200 (Ok), 201 (Created), 400 (Bad Request), 404 (Not Found) [31].

Päringute tegemiseks on klientrakenduse poolel võetud kasutusele axios teek, mis pakub meetodeid, mis vastavad kõikidele vajaminevatele HTTP verbidele. Axios teisendab JSON andmed automaatselt, seega puudub vajadus kirjutada käsitsi koodi andmete teisendamiseks ühest vormingust teise [32].

4.2 Klientrakenduse arhitektuur

Põhiliseks kaustaks on src, mille sisu jaguneb omakorda erinevateks kaustadeks:

- **Components** – sisaldab taaskasutatavaid komponente ehk koodiblokke, mida saab erinevates vaadetes kasutada, et vähendada korduvate koodiridade arvu.
- **Router** – sisaldab loogikat määramaks, millist vaadet vastavalt teekonnale laadida. Erinevate lehtede vahel navigeerimine on lahendatud dünaamiliselt, mis tähendab, et lehele vastav teekond sisaldab muutuvat lüli, näiteks plaani identifikaatorit.
- **Styles** – sisaldab rakenduse globaalset CSS koodi ehk stiili, mis rakendub kõikidele rakenduse lehtedel üheselt. Sel viisil on tagantjärele lihtsam visuaaliga seotud muudatusi läbi viia. Igat faili eraldi muutma ei pea, kui on soov näiteks teksti värvi või nupu kujundust muuta. Sel viisil saab tagada, et kogu rakenduses säilib ühtne stiil.
- **Views** – sisaldab faile, mis esindavad ühte kindlat vaadet ehk lehte rakenduses. Iga fail koosneb peamiselt kahest osast, milleks on template, mis kirjeldab lehe struktuuri ja sisu ja script, mis sisaldab andmeid ja kirjeldab lehe käitumise erinevate meetoditega.

4.3 Serverirakenduse arhitektuur

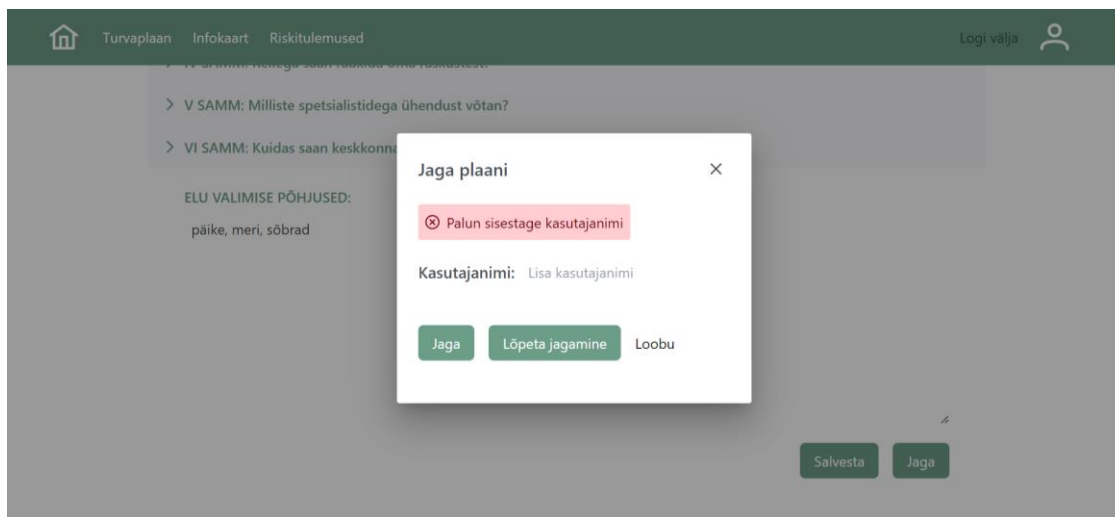
Struktuuriüksuste eraldamine ning koodi organiseerimine on saavutatud kaustade kasutamisega. Eraldatud kaustad hoiavad endas kindlat vastutust. Kuna rakenduse edasiarenduseks peab jääma hea võimalus ja selle käigus võib koodibaas tunduvalt suurened, siis kaustade kasutamine aitab seda edaspidi hõlpsamini hallata. Koodi organiseerimine kihtidesse annab võimaluse madala taseme funktsionaalsuse kasutamiseks kogu rakenduses, sel juhul on vaja kirjutada vähem koodi, samuti aitab järgida “ära korda ennast” ehk DRY printsiipi [33]. Kihilise arhitektuuriga rakendustes saab kehtestada piiranguid selle kohta, millised kihid võivad suhelda teiste kihtidega ja kui ühes kihis muudatusi tehakse, peaksid mõjutatud olema ainult need kihid, mis seda kasutavad [33]. Piirates sõltuvust teistest kihtidest, saab üksiku muudatuse mõju leevendada nõnda, et see ei mõjutaks kogu rakendust [33]. Loogika organiseerimine erinevatesse kihtidesse on levinud tehnika koodi organiseerituse parandamiseks [33]. Käesoleva lõputöö serverirakenduses on loodud järgmised kihid:

- **Models** – sisaldab klasse, mis illustreerivad erinevaid andmeid, mida rakendus haldab.
- **Data** – sisaldab DataContext klassi, mis pärib Entity Framework Core DbContext klassi. Entity Framework Core on ORM, mis kombineerib objektorienteeritud programmeerimise relatsioonilise andmebaasiga ja aitab lahendada kokkusobimatuse probleemi C# klasside ja andmebaasi struktuur vahel [10]. DataContext klass haldab andmebaasiühendust ning seda kasutatakse andmete kätte saamiseks ja salvestamiseks [34]. Samas klassis on määratud ka andmeklasside ja tabelite vahelised seosed, näiteks üks-ühele seos turvaplaani ja infokaardi vahel või üks-mitmele seos turvaplaani ja riskitulemuse vahel.
- **Controllers** – sisaldab klasse, mis võtavad vastu päringuid ja suhtlevad Services kihiga andmete hankimiseks ja manipuleerimiseks.
- **Services** – sisaldab rakenduse ärioloogikat.

4.4 Veebirakenduse disain

Veebirakenduses kasutatud värvilahendus on eesmärgipäraselt roheline. Psühholoog Toivo Niibergeri [35] sõnade kohaselt mõjutavad värvid inimese vaimset ja füüsilist tervist. Roheline on rahustav ja sisemist tasakaalu toetav ning toniseeriv värv. Psühholoogiliselt põhjustab roheline puhtuse, värskuse ja uuennemistunde. Näiteks ka rohelised ruumid sobivad eriti hästi südamehäirete korral ja kõrgvererõhutõbe põdejatele. Roheline laiendab ka kapillaare ja tekitab kehas soojatunde, leevendab ärritust, samuti virgutab aju ning ehitab üles lihaseid ja kudesid. Roheline reguleerib tundeelu ja tugevdab väsimusest, haigustest või negatiivsetest tunnetest väsitatud keha [35].

Ka rakenduse kujunduse loomisel on silmas peetud selle lõppkasutajaid, sest kõrgendatud haavatavuse seisundis inimesed vajavad lihtsaid lahendusi. Veebirakendus on tehtud piisavalt lihtne, et ei pea aega kulutama arusaamaks, kuidas seda kasutada. Kui kasutaja midagi valesti teeb, siis tekivad ekraanile suunavad või hoiatavad sõnumid, mida illustreerib joonis 5. Seal on näha punasega kuvatud sõnumit, mis on tekkinud pärast jagamisnupu vajutamist seetõttu, et lahter nõustaja kasutajanime sisestamiseks jäeti tühjaks.



Joonis 5. Vea sõnum plaani jagamisel

Segaduse vältimiseks kuvatakse navigeerimisribal olevad lingid vastavalt kasutaja rollile. Kliendil pole vaja jagatud plaanide vaadet, ning seetõttu on see näha ainult nõustaja rollis olevale veebirakenduse kasutajale (joonis 5). Eeldusel, et nõustajad kasutavad rakendust siiski ainult teiste abistamiseks, ei kuvata neile linke oma personaalsele turvaplaanile, riskitulemuste lehele ja infokaardile (joonis 6).



Joonis 6. Jagatud plaanide vaade

Joonis 6 illustreerib lõpplahendust jagatud plaanide vaatele, mis erineb veidi selle algsest ideest ja skeemist. Nõustajal on võimalus otse jagatud plaanide vaatest kliendi turvaplaani, infokaardi või riskitulemuste lehele saada ning ei pea nende linkide jaoks eraldi lehte avama nagu oli selle vaate esialgne idee. Võrreldes algsete skeemidega on kaotatud ka nupp turvaplaanide loomiseks. Seda pole tegelikult vaja, sest uue kasutaja loomisel, luuakse ka juba uus turvaplaan.

Samuti on pööratud tähelepanu disaini responsiivsusele, et rakendust saaks erinevates seadmetes mugavalt kasutada. Veebirakenduse kujundus kohandub vastavalt ekraani suurusele nagu on näha joonisel 7, kus navigeerimisribal olevad lingid saab väiksema ekraani korral nupu alt avada, samuti paigutub plaan ekraani ühest servast teise, mitte selle keskele. Selline lähenemisviis on saavutatud kasutades CSS @media reeglit, millega on näiteks teksti suurust ja ääriselise mõõtmeid väiksemate ekraanide puhuks kahandatud [36].



Joonis 7. Turvaplaani vaade nutitelefonist

4.5 Ärijuhtumi sisend

Eestis läbiviidud uuringu [37] kohaselt on sõltumata viisist, kuidas suitsiidiriski hinnatakse, väga oluline tagada suitsiidse inimese ohutusplaan (enesetapukriisi turvaplaan konkreetsele inimesele edaspidi sarnaste juhtumite ennetamiseks). Eestis kasutatakse turvaplaani näiteks Tallinna Lastehaigla Laste Vaimse Tervise Keskuses. Uuringu kohaselt vaimse tervise valdkonnas regulatsioonidest ja tegevusjuhistest suitsiidse inimese käsitlemiseks ja suitsiidiriski hindamiseks palju ei teata ning neist tuntakse puudust. Puudust tuntakse eelkõige lühikestest, konkreetsetest ja praktilistest juhistest. Kuigi uuringus osalejad leidsid, et infot, näiteks koolituste, regulatsioonide ja

testide kohta on soovi korral võimalik internetist leida ja see on oskuslikele otsijatele kättesaadav, oleks siiski vajalik ühte konkreetset veebikeskkonda, kuhu koonduks kogu usaldusväärne ja regulaarselt uuendatav info nii koolituste, juhendite kui ka Eestis kasutusel olevate suitsiidiriski hindamisvahendite kohta. Selles veebikeskkonnas peaks inimene saama vajadusel ise teste eeltäita ja ennast regulaarselt skriinida. Ohutusplaani koostamine suitsiidse inimese abistamiseks peab olema üks osa suitsiidiriski hindamise protsessist ning selle käigus tuleb suitsiidse inimesega läbi rääkida ja kirja panna vähemalt järgmine – millised on konkreetset tema puhul varajased suitsiidikäitumise ohumärgid (emotsionaalsed, tunnetuslikud, käitumuslikud), millised on enesehoole ja suitsiidse impulsi maandamise soovitusel, millised on abivõimalused (olulised lähedased inimesed ja spetsialistid koos kontaktandmetega) [37].

Käesoleva lõputöö raames loodav enesetapukriisi turvaplaan oleks üks osa veebikeskkonnast, mis koondab tervikliku info Eestis kasutusel olevatest suitsiidiriski hindamisvahendite, juhendite ja ennetava materjali kohta.

Enesetapukriisi turvaplaani toel enesetapuprotsessi sekkumine on enesetappude ennetamise kaasaegses praktikas ja uuringutes levinud soovitus. Ülevaateuuringud [38] näitavad järjepidevalt selle sekkumise tõhusust. Turvaplaan on tõhus, madala intensiivsusega ja paindlik sekkumisvahend suitsiidiriskis inimese toetamiseks.

Stanley & Brown [39] uuringu kohaselt on turvaplaan tõhus sekkumisvahend täiskasvanutele ja lastele. Kvantitatiivsed uurimistulemused näitavad turvaplaani sekkumise mõju enesetapumõtete ning suitsidaalse käitumise vähenemisele, depressiooni ja lootusetuse tunde vähenemisele; haiglaravi vajaduse vähenemisele; ravis osalemise paranemisele. Kvalitatiivsed uuringud näitavad, et turvaplaani sekkumine on kasutatav nii digitaalselt kui paberkandjal, näost-näku sekkumisel ja veebirakendusena kas iseseisvalt või toetaja juhendamisel täidetav ning rakendatav nii eraldiseisvana kui kombineeritud sekkumisena [39].

Rainbow jt [40] uurisid turvaplaani rakenduse kasutamist kolme kuu perioodi jooksul. Tulemused näitasid enesetapumõtete märkimisväärset vähenemist ja enesetapumõtetega toimetuleku suurenemist. Kõrgem enesetapumõtetega toimetulek näitab enesetapumõtete vähenemist. Suurem tajutav kasulikkus ja isikupärastatud sisu kolme kuu jooksul olid seotud suurema enesetapuga seotud toimetulekuga. Autorid järeldavad, et praktikud

peaksid andma klientidele võimaluse luua ohutusplaane isikupärastatud (mitte üldiste) strateegiatega, mida klient peab kasulikuks. Sellised plaanid võivad tugevdada uskumusi enesetapumõtete toimetuleku kohta, mis omakorda vähendab aja jooksul enesetapumõtteid [40].

Sekkumise läbiviimise ajal saab toetaja jagada infot enesetapukäitumise ja selle kujunemise kohta ning avada riskitegureid. Suitsidaalne inimene jagab oma lugu ja vaimse tervise spetsialist koos kliendiga tuvastab kriisi riskitegurid ning ennetavad toimingud. Koostöös koostatakse kirjalik, prioriseeritud nimekiri konkreetsetest toimetulekustrateegiast ja toetusallikatest, mida saaks kasutada tulevase suitsidaalse kriisi leevendamiseks [39].

Enne ohutusplaani koostamist on vajalik hinnata enesetapu riskitegurid ning selleks on vajalik infokaardi täitmine. Suitsiidiriski varajane märkamine ja hindamine on oluline enesetappude ennetamise osa [37]. Infokaart on mõeldud spetsialistile täitmiseks, sest sisaldab teaduslikult tõestatud enesetapuriski komponente mis mõjutavad enesetapuriski omavahelises koostoimes. Seetõttu on nende tõlgendamiseks vajalik vaimse tervise valdkonna ettevalmistus ning ei piisa igapäevateadmised ja infokaart on muudetav vaid spetsialisti vaatest.

Allpool on toodud ohutusplaani sammud, mis põhinevad Stanley ja Brown [39] uuringul ning lepatakse kokku koostöös suitsidaalse inimesega:

- Kriisile eelnevate ohumärkide tuvastamine (mõtted, tuju, olukorrad, käitumisviisid, ettekujutused)
- Tegevused, mida inimene ise saab teha, et oma enesetunnet parandada
- Kellega saab rääkida/kuhu minna, et oma tähelepanu kõrvale juhtida?
- Kellega saab rääkida oma raskustest?
- Milliste spetsialistidega ühendust võtta?
- Kuidas saab oma keskkonna turvalisemaks muuta?
- Elu valimise põhjused – mis teeb elu elamisväärseks? [39]

5 Järeldused

Arenduse käigus sai autor regulaarselt tagasisidet ohvriabitöötajate meeskonnajuhilt. Seetõttu saab kindlalt väita, et rakendus sisaldab kõiki osi, mida nõustaja kriisis isikule abi osutamiseks päriselt vajab. Enesetapukriisi turvaplaani rakendus on vaimse tervise spetsialisti tööd oluliselt lihtsustav abivahend. Rakendus võimaldab ajasäästlikult ja kaugnõustamise teel luua kriisis klientidele personaalne turvaplaan enesetapumõtete või -käitumise korral.

Enesetapumõtetega klientide puhul on kõige olulisem abi kiire ja lihtne kättesaadavus. Rakendus tagab vaimse tervise spetsialistile ja kliendile juurdepääsu enesetapukriisi turvaplaanile, olles kättesaadav igal ajal ja igas kohas. Tavapäraselt on nutitelefon inimesel igal pool kaasas ja visuaali responsiivsuse lahendamisele saab seda telefonist avada. See võimaldab kokku hoida nii spetsialisti kui ka kliendi aega. Nõustajatel kulub näiteks vähem aega riski hindamisele, sest punkte pole vaja ise kokku lugeda, vaid selle teeb rakendus spetsialisti eest ära. Kuna riski hindamine ja infokaart on paigutatud ühele lehele, on spetsialistil kergem mõlemat korraga täita, kui selleks peaks vajadus tekkima. Ka soodustab rakendus koostöövõimalust erinevate spetsialistidega. Kui klient soovib oma turvaplaani jagada mitme erineva nõustajaga, ei jää igaühele oma personaalne koopia kliendi plaanist. Sel viisil on tagatud, et relevantsem info jõuab kõigini, kui kaasatud on rohkem kui üks spetsialist. Ohvriabitöötajad on liinil vahetustega ja on suur tõenäosus, et sama kliendi kõne saab igal korral erinev nõustaja.

Lisaks aitab rakendus jõuda rohkemate abivajajateni. Seoses majanduslikult keerulise olukorraga Eestis üldisemalt, ei pruugi olla võimalust ka spetsialiste juurde värvata. Samas abivajajate arv ajas kasvab. Rakenduse kasutamine võimaldab tõhusamalt hallata kliente ning jõuda tegelemiseni kõigi pöördujatega.

Enesetapukriisi turvaplaan võimaldab lihtsalt ja mugavalt jälgida kliendi progressi ja plaani kasutamissagedust ning seeläbi vajadusel sekkuda. Rakendus aitab kliente individuaalselt toetada ja juhendada nende enesetapumõtete või -käitumise korral.

Arenduse käigus sai läbi viidud ka refaktoreerimine, et jääks hea võimalus edasiarenduseks. Pöörati tähelepanu koodi loetavusele järgides parimaid praktikaid näiteks meetodite nimetamisel.

6 Võimalused edasiarenduseks

Üheks arenduskohaks on turvalisus ja jõudlus. Kuna tegemist on rakendusega, mis hoiab väga tundlikku informatsiooni iga kliendi kohta, peaks puuduma ligipääs kõrvaliste isikute ressurssidele. Rakendus vajab täiendavaid kontrole ja meetodeid, mis ligipääsu piiraksid. Hetkel on kasutusel HTTP, sest rakenduse arenduse käigus kasutati testandmeid. Kui aga rakendus saab põhjalikult testitud, puudused kõrvaldatud ning seejärel avalikkusele üles pandud, võetakse HTTP asemel kindlasti kasutusele HTTPS, mis on vajalik turvalisuse tõstmiseks. HTTPSi kasutuselevõtuks on vaja soetada allkirjastatud sertifikaat. Rakenduse toimimist saaks samuti sujuvamaks muuta. Peale sisselogimist jõuavad andmed lehele aeglaselt, selleks on vaja lehte värskendada ja teha mõni navigeerimine enne, kui vaade on ootuspärane. Kasutajasõbralikkus on iga rakenduse puhul oluline aspekt, seega tuleks sellele edasiarenduse käigus rohkem tähelepanu pöörata. Ka turvalisuse osa lahendades vähenes rakenduse jõudlus tunduvalt, aga nende teostamine ei tohiks ideaalis teineteise ootuskohast funktsioneerimist välistada. Samuti on veel võimalusi koodi edasiseks refaktoreerimiseks ja struktureerimiseks.

7 Kokkuvõte

Eesti enesetapunäitajad on teiste Euroopa riikidega võrreldes väga kõrged ja samal ajal on puudus abi pakkumiseks sobivatest eestikeelsetest vahenditest.

Enesetapukriisi turvaplaan on maailmas levinud ja tõhus vaimse tervise valdkonna töövahend. Turvaplaan toetab suitsidaalse inimese enesejuhtimist ning pakub tuge enesetapumõtetega toimetulekuks. Seni eksisteerivate lahenduste suurimaks puudujäägiks on mugava koostöövõimaluse puudumine. Ei saa jagada plaani viisil, et ka teised, näiteks nõustajad saaksid üksikut plaani muuta ja selle loomise kulgu lihtsasti jälgida saaks. Praegu eksisteerivad pigem sellised lahendused, kus turvaplaani saab muuta ainult üks inimene. Samuti on puuduseks kehv kättesaadavus. Oma plaanile ei saa erinevatest brauseritest ja seadmetest ligi viisil, et seda ka edasi saaks arendada.

Bakalaureusetöö eesmärgiks oli luua veebirakendus, mis võimaldaks kliendi ja spetsialisti koosarendatava turvaplaani loomist.

Eesmärgi saavutamiseks viidi läbi küsitlus tellijaga, analüüsiti temaatilist kirjandust, probleemi ja alternatiivseid lahendusi. Nende tegevuste põhjal selgitati välja, milline võiks olla uus lahendus probleemile. Seejärel valiti parimad tööriistad ja vastavalt nõuetele loodi rakendus, mis läbis testimise ja mille tulemusi analüüsiti. Enesetapumõtetega klientide puhul on kõige olulisem abi kiire ja lihtne kättesaadavus. Rakendus tagab vaimse tervise spetsialistile ja kliendile juurdepääsu enesetapukriisi turvaplaanile, olles kättesaadav igal ajal ja igas kohas.

Bakalaureusetöö eesmärk sai täidetud - loodud enesetapukriisi turvaplaan võimaldab lihtsalt ja mugavalt jälgida kliendi progressi ja plaani kasutamissagedust ning seeläbi vajadusel operatiivselt sekkuda enesetapukriisi süvenemisesse. Rakendus võimaldab kliente individuaalselt toetada ja juhendada nende enesetapumõtete või -käitumise korral.

Kasutatud kirjandus

- [1] K. Aavik ja L.-M. Väli, „Pealinn.ee“, Ajaleht Pealinn, 11 mai 2023. [Võrgumaterjal]. Available: <https://pealinn.ee/2023/05/11/eesti-enesetapunaitajad-on-uhed-euroopa-suurimad-suurimas-ohus-on-mehed-aga-muret-teeb-ka-alaealiste-suitsiidikatsete-kasv/>. [Kasutatud 26 veebruar 2024].
- [2] V. Ojaperv, „Tartu Postimees“, Postimees, 10 september 2023. [Võrgumaterjal]. Available: <https://tartu.postimees.ee/7851983/suitsiidiennetuse-paeval-raagitakse-ka-nendest-kes-peale-lahedase-kaotust-edasi-elavad>. [Kasutatud 26 veebruar 2024].
- [3] Organisation, World Health, ENESETAPPUDE ENNETAMINE: ABIKS ESMATASANDI TERVISHOIUTÖÖTAJATELE, Tallinn: Eesti-Rootsi Vaimse Tervise ja Suitsidoloogia Instituut, 2008.
- [4] Geenius Meedia OÜ, „TerviseGeenius“, Geenius Meedia OÜ, 10 oktoober 2022. [Võrgumaterjal]. Available: <https://tervise.geenius.ee/rubriik/vaimne-tervis/ott-oja-igal-aastal-jaab-eestis-200-000-inimest-vaimse-tervise-abita/>. [Kasutatud 26 veebruar 2024].
- [5] Lõuna Ekspress OÜ, „Lõunaestlane“, Lõuna Ekspress OÜ, 9 september 2022. [Võrgumaterjal]. Available: <https://lounaestlane.ee/suitsiidi-tottu-suri-eelmisel-aastal-kolm-korda-rohkem-inimesi-kui-liiklusonnetustes/>. [Kasutatud 26 veebruar 2024].
- [6] CENTRE FOR SUICIDE PREVENTION, „CENTRE FOR SUICIDE PREVENTION“, Canadian Mental Health Association, 9 september 2019. [Võrgumaterjal]. Available: https://www.suicideinfo.ca/local_resource/safety-plans/. [Kasutatud 26 veebruar 2024].
- [7] K. Jõe, „Vaimse tervise toetamise veebirakenduse IntroScope prototüüp“, Tallinna Tehnikaülikool, Tallinn, 2022.
- [8] K. Hansen, „Üksildust leevendav eneseabirakendus“, Tallinna tehnikaülikool, Tallinn, 2020.
- [9] Ohvriabi, „PalunAbi“, Sotsiaalkindlustusamet, [Võrgumaterjal]. Available: <https://www.palunabi.ee/et/ohvriabi>. [Kasutatud 27 veebruar 2024].
- [10] A. F. M. Cardoso, Enterprise Applications with C# and .NET, London: BPB Online, 2023.
- [11] M. O. Melnikov, D. V. Kornienko ja S. V. Mishina, „The Single Page Application architecture when developing secure Web services“, *Journal of Physics: Conference Series*, p. 3, 26 oktoober 2021.
- [12] R. Vyas, „Comparative Analysis on Front-End Frameworks for Web Applications“, *International Journal for Research in Applied Science & Engineering Technology*, kd. 10, nr 7, pp. 298-299, 2022.

- [13] N. Kwartalnyi, „Inoxoft,“ Inoxoft, 20 jaanuar 2023. [Võrgumaterjal]. Available: <https://inoxoft.com/blog/web-application-architecture-components-models-and-types/>. [Kasutatud 29 veebruar 2024].
- [14] N. Ferguson, „CareerFoundry,“ CareerFoundry, 6 veebruar 2024. [Võrgumaterjal]. Available: <https://careerfoundry.com/en/blog/web-development/whats-the-difference-between-frontend-and-backend/>. [Kasutatud 29 veebruar 2024].
- [15] A. Annamaa, „Programmeerimise Õpik,“ Tartu Ülikooli Arvutiteaduse instituut, 5 detsember 2017. [Võrgumaterjal]. Available: https://progeopik.cs.ut.ee/programmidevaheline_suhtlus.html. [Kasutatud 29 veebruar 2024].
- [16] The Postman Team, „Postman,“ 28 juuni 2023. [Võrgumaterjal]. Available: <https://blog.postman.com/rest-api-examples/>. [Kasutatud 2 veebruar 2024].
- [17] N. Hekman, „Youtube,“ IBM Technology, 23 oktoober 2020. [Võrgumaterjal]. Available: <https://www.youtube.com/watch?v=lsMQRaeKNDk>. [Kasutatud 2 märts 2024].
- [18] A. Lock, ASP.NET Core in Action, Third Edition, New York: Manning Publications Co., 2023.
- [19] Ruby, „Ruby,“ Ruby, [Võrgumaterjal]. Available: <https://www.ruby-lang.org/en/about/>. [Kasutatud 10 mai 2024].
- [20] S. Pramanick, „Kellton,“ Kellton, 1 veebruar 2024. [Võrgumaterjal]. Available: <https://www.kellton.com/kellton-tech-blog/best-backend-web-development-frameworks>. [Kasutatud 28 aprill 2024].
- [21] P. McFedries, HTML, CSS, & JavaScript All-in-One For Dummies, New Jersey: John Wiley & Sons, Inc., 2023.
- [22] C. Nance, TypeScript Essentials, Birmingham: Packt Publishing Ltd., 2014.
- [23] M. Ernst, „Version control concepts and best practices,“ 4 veebruar 2024. [Võrgumaterjal]. Available: https://homes.cs.washington.edu/~mernst/advice/version-control.html#Introduction_to_version_control. [Kasutatud 29 veebruar 2024].
- [24] GitLab, „GitLab,“ GitLab, [Võrgumaterjal]. Available: <https://about.gitlab.com/topics/version-control/>. [Kasutatud 2 mai 2024].
- [25] Azure, „Azure,“ Azure, [Võrgumaterjal]. Available: https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-are-databases?fbclid=IwAR16enbci5puMV-HZYP7Hoq3cvusRM7iKdLfj_Nrqz3FvD4pvjLIDkVkOG8. [Kasutatud 8 märts 2024].
- [26] V. Le, „Orient Software,“ Orient Software, 6 juuli 2023. [Võrgumaterjal]. Available: <https://www.orientsoftware.com/blog/database-web-applications/>. [Kasutatud 10 mai 2024].
- [27] Aeris Consulting, „Docker The Compact Container Platform,“ [Võrgumaterjal]. Available: <https://www.aeris-consulting.com/wp-content/uploads/2022/04/Docker.pdf>. [Kasutatud 7 mai 2024].
- [28] T. Charboneau, „Docker,“ Docker, 5 oktoober 2022. [Võrgumaterjal]. Available: <https://www.docker.com/blog/how-to-use-the-postgres-docker-official-image/>. [Kasutatud 27 aprill 2024].

- [29] new target, „new target,“ new target, 26 mai 2020. [Võrgumaterjal]. Available: <https://www.newtarget.com/web-insights-blog/separation-of-concerns-in-web-design-and-development/>. [Kasutatud 22 aprill 2024].
- [30] MDN, „MDN Web Docs,“ Mozilla, 15 märts 2024. [Võrgumaterjal]. Available: https://developer.mozilla.org/en-US/docs/Learn/Server-side/First_steps/Client-Server_overview. [Kasutatud 22 aprill 2024].
- [31] M. Pattankar ja M. Hurbuns, *Mastering ASP.NET Web API*, Birmingham: Packt Publishing Ltd., 2017.
- [32] F. Copes, „freeCodeCamp,“ freeCodeCamp, 11 aprill 2018. [Võrgumaterjal]. Available: <https://www.freecodecamp.org/news/simple-http-requests-in-javascript-using-axios-272e1ac4a916/>. [Kasutatud 21 04 2024].
- [33] S. “. Smith, „Architecting Modern Web Applications with ASP.NET Core and Microsoft Azure,“ Microsoft Developer Division, Washington, 2023.
- [34] Pragimtech, „Pragim Technologies,“ Pragimtech, [Võrgumaterjal]. Available: https://www.pragimtech.com/blog/blazor/asp.net-core-rest-api-dbcontext/#google_vignette. [Kasutatud 27 aprill 2024].
- [35] T. Niiberg, *Ela ja küsi! 500 elulist küsimust*, Goodnews, 2021.
- [36] W3Schools, „W3Schools,“ W3Schools, [Võrgumaterjal]. Available: https://www.w3schools.com/cssref/css3_pr_mediaquery.php. [Kasutatud 27 aprill 2024].
- [37] M. Sisask, K. Kriisk, M. Medar, L. Zinoveva, O. Lupanova, A. Samuel, H. Väärä, O. Käärman, K. Mikiver, A. Värnik, P. Värnik, R. Mere, K. Mikiver ja H. Värnik, „Suitsiidse patsiendi käsitus esmatasandi tervishoius (sh kooli- ja töötervishoid), kiirabis ja erakorralise meditsiini osakondades (SUIPA),“ Tallinna Ülikool ja ERSI, Tallinn, 2023.
- [38] P. Crowley, D. Carmichael, S. Aryobi, R. A. Goldszmidt, R. Isard, C. Easton, J. Holmes ja S. Murphy, „Suicide Safety Planning: A Systematic Review Supporting OT Research & Practice,“ *The American Journal of Occupational Therapy*, kd. 77, nr Supplement 2, 2023.
- [39] B. Stanley ja G. K. Brown, „Safety Planning Intervention: A Brief Intervention to Mitigate Suicide Risk,“ *Cognitive and Behavioral Practice*, kd. 19, nr 2, pp. 256-264, 2012.
- [40] C. Rainbow, R. Tatnell, G. Blashki ja G. A. Melvin, „Perceived usefulness of self-guided versus collaborative suicide safety plans in online help-seekers.,“ *Crisis: The Journal of Crisis Intervention and Suicide Prevention*, 2024.
- [41] L. S. Sterling, *The Art of Agent-Oriented Modeling*, London: The MIT Press, 2009.

Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks¹

Mina, Eliise Tall

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Veebirakendus enesetapukriisi ennetamiseks“, mille juhendaja on Meelis Antoi
 - 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.

15.05.2024

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 loomingulise 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.