

Tallinna Ülikool
Digitehnoloogiaste Instituut

JUHUSLIK – MEELELAHUTUSRAKENDUSE PROTOTÜÜP

Bakalaureusetöö

Autor: Kevin Rull
Juhendaja: Jaagup Kippar

Autor: „ „ 2016
Juhendaja: „ „ 2016
Instituudi direktor: „ „ 2016

Tallinn 2016

Autorideklaratsioon

Deklareerin, et käesolev seminaritöö on minu töö tulemus ja seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

.....

(kuupäev)

.....

(autor)

Lihlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina _____ (sünnikuupäev: _____)
(*autori nimi*)

1. annan Tallinna Ülikoolile tasuta loa (lihlitsentsi) enda loodud teose

(*lõputöö pealkiri*)

mille juhendaja on _____,
(*juhendaja nimi*)

säilitamiseks ja üldsusele kättesaadavaks tegemiseks Tallinna Ülikooli Akadeemilise Raamatukogu repositooriumis.

2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.
3. kinnitan, et lihlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tallinnas/Haapsalus/Rakveres/Helsingis, _____
(*digitaalne*) allkiri ja kuupäev

Sisukord

MÕISTETE LOETELU	5
SISSEJUHATUS	6
1. RAKENDUSE VÕRDLUS TURUL OLEVATE SARNASTE RAKENDUSTEGA	8
1.1. RANDO – RAKENDUS AINULT IOS SEADMETEL	8
1.2. 9GAG – SUURIM MEELELAHUTUSVEEB.....	10
1.3. JUHUSLIK – MEELELAHUTUSRAKENDUSE PROTOTÜÜP.....	12
2. RAKENDUSE ARENDUS	14
2.1. ESIALGNE RAKENDUSE PLAAN	14
2.2. RAKENDUSE ARENDUSE KÄIK.....	16
2.2.1. <i>Rakenduses kasutatavad API-d</i>	16
2.2.2. <i>Rakenduses kasutatav muu sisu</i>	18
2.2.3. <i>Rakenduse funktsionaalsuse arendus</i>	19
2.2.4. <i>Rakenduse andmebaas</i>	21
2.2.5. <i>Rakenduse struktuur</i>	24
3. TESTKASUTAJATE TAGASISIDE NING VEEBILEHEL KÄITUMISE ANALÜÜS	27
3.1. TAGASISIDE ANALÜÜS	27
3.2. TESTKASUTAJA VEEBILEHEL KÄITUMISE ANALÜÜS	30
KOKKUVÕTE	32
KASUTATUD KIRJANDUS	33
SUMMARY.....	35
LISAD.....	36

Mõistete loetelu

API – (Application Programming Interface). Programmiliides, mille alusel rakendusprogramm kasutab teise rakendusprogrammi teenuseid. (<http://www.vallaste.ee>)

cURL – avatud lähtekoodiga käsurea tööriist, millega on võimalik andmeid vahetada URL süntaksiga, toetades mitmeid protokolle. (<https://curl.haxx.se/>)

MySQL – Väga populaarne avatud lähtekoodiga relatsioonbaasihaldur. Kasutab struktureeritud päringukeelt SQL. (<http://www.vallaste.ee>)

SDK – (Software Development Kit). Arendustarkvara programmipakett, mis võimaldab programmeerijal luua rakendusi konkreetsele platvormile. (<http://www.vallaste.ee>)

SQL Injection – Tehnika, mida kasutatakse SQL päringute muutmiseks, veebilehe turvalisuse ohustamiseks ja ka andmebaasis muudatuste tegemiseks läbi veebilehel kuvatate vormide kaudu. (http://www.w3schools.com/sql/sql_injection.asp)

GIF – Joonisfilmile sarnanev mitmest pidevalt korduvast kaadrist koosnev GIF-vorming. (<http://www.vallaste.ee>)

PHP – (Hypertext Preprocessor). PHP on platvormist sõltumatu skriptikeel ja kujutab endast alternatiivi Microsoft'i ASP keelele.

HTML – (Hypertext Markup Language). Hüpertekst-märgistuskeel on enamlevinud kodeerimissüsteem veebidokumentide loomiseks. (<http://www.vallaste.ee>)

Javascript – Netscape'i poolt välja töötatud skriptikeel, mis võimaldab veebis luua interaktiivseid ja dünaamilisi veebisaitte. (<http://www.vallaste.ee>)

jQuery – Võimalusterohke Javascripti teek, mis oma API'ga teeb HTML'is sisu manipuleerimise, muutmise, animeerimise veelgi lihtsamaks. (<https://jquery.com/>)

Bootstrap – Bootstrap on stiilne, intuitiivne ja võimas front-end raamistik kiiremaks ja lihtsamaks veebi arenduseks. (<https://github.com/twbs/bootstrap>)

Regex – (Regular Expression). Regulaaravaldis on string, mis kirjeldab või langeb kokku mingi stringide hulgaga vastavalt kindlatele süntaksireeglitele. (<http://www.vallaste.ee>)

Endpoint – Mingi teatud protsessi lõpptulemus. (<http://www.vallaste.ee>)

JSON – (JavaScript Object Notation). On kerge andmete vahetuse formaat, mida suudavad inimesed kergelt lugeda ja kirjutada ning arvutitel lihtne formaati genereerida ja sõeluda. (<http://www.json.org/>)

Sissejuhatus

Bakalaureusetöö teemaks on autor valinud meelelahutusrakenduse prototüübi loomise. Rakenduse esialgne nimi Juhuslik tuleneb inglise keelsest sõnast *Random*. Rakendus on avalik veebiaadressil <http://clickndrive.ee/juhuslik>.

Autori töö teema valiku põhjendus seisneb selles, et tuua inimestele meelelahutust natukene uuemas võtmes ning töö autoril oleks huvitav rakenduse prototüüpi arendada. Rakenduse idee sai otseselt alguse ühel suveõhtul kolleegide koosviibimisel. Esialgu oli rakenduse idee väga lihtne ning ka üksluine, kuid pika ajurünnaku tulemusena saadi rakendusele mitmeid lisafunktsioone juurde, millest kindlasti tuleb ka juttu käesolevas töös. Töö autor peab õigeks ka ära mainida, et ta ei ole arendatava rakenduse idee üksainus autor.

Töö teema aktuaalsus seisneb selles, et seostada rakenduse kasutajat käituma nii nagu ta käituks realselt tänaval. See tähendab seda, et tänaval käies inimene ei juhi enda ümber toimuvat, vaid ainult seda, mida tema ise teeb. Tuues eelnev näide rakendusse, siis kasutajal ei ole võimalik kontrollida rakenduse sisu, kuid tal on võimalik kontrollida, kas ta soovib jagada seda sisu või näiteks midagi hoopis lisada sisusse.

Töö teema on uudne selle poolest, et hetkel ei leidu nii juhusliku sisuga rakendust ja funktsionaalsusega, mis on kõik ühes rakenduses koos. Autor soovib rakenduse kasutaja tuua reaalsusesse, kus kasutaja loodetavasti taipab üsna ruttu, et tema kontrollida ei ole see, mis toimub rakenduse sees.

Töö eesmärk on luua rakenduse prototüüp, mis võiks kinnitada ja ka selgitada rakenduse funktsionaalsust ning lähtuvalt testgrupi tagasisidest teha järeldusi, kas on üldse taolisel rakendusel kohta tänapäevases nutimaailmas.

Töö on jaotatud kolme osa vahel. Esimeses osas töö autor toob välja võimalikult sarnaseid olemasolevaid rakendusi ning võrdleb enda loodavat rakendust olemasolevatega. Teises osas kirjeldab autor rakenduses kasutatavaid süsteeme ning kasutajate võimalusi. Kolmandas osas teeb autor järeldused testgrupi tagasiside põhjal ja nende käitumisest rakenduses.

Rakenduse arendatakse veebilehele. Tagasüsteemide poolt kannab hoolt PHP ning kujunduse ja eesprogrammidenä kasutatakse HTML, Javascripti, JQuery ja Bootstrapi, kuigi kujundusele erilist rõhku ei panda. Rakenduse prototüüpi testitakse testgrupis.

1. Rakenduse võrdlus turul olevate sarnaste rakendustega

Tänapäeval on internet täis igasuguseid naljalehti nagu näiteks 9GAG, mis on järgnevalt loetletutest kõige suurema külastatavusega viimase 6 kuu jooksul, 292 miljonit külastust.¹ Lisaks 9gag lehele leiab ka veebist sellised saidid nagu 4chan, Imgur, reddit ja palju teisi lehti, mis sisaldavad sisu, mis on mõeldud külastajale meelelahutuseks. Käesolev peatükk toob välja teised rakendused, mida võib pidada sarnaseks arendatava rakenduse prototüübiga. Autor võrdles rakendusi, kasutades selleks vaid nutitelefone.

1.1. Rando – rakendus ainult iOS seadmetel

Hiljuti välja tulnud rakendus Rando, esmane versioon publitseeriti 03.03.2016, on saadaval ainult iOS seadmetele ning rakenduse arendaja on mees nimega David Barnard. Tegemist on äppiga, mille kaudu on kasutajal võimalik jagada juhuslikult saadud pilti või tsitaati.

Rakendus on oma väljanägemiselt väga pilkupäädev ning hästi disainitud. Kogu rakenduse kasutamine on kasutajale võimalikult lihtsaks tehtud. Põhimõtteliselt on võimalik genereerida ja jagada oma suvaline sisu kolme nupuvajutusega (vt joonis 1). Rakendust avades kuvatakse kolme peamist valikut: Photo, Gif, Quote. Nende valikute vahel on võimalik valida, millisest kategooriast soovib kasutaja suvalist sisu endale kuvada. Esimeseks sammuks on kasutajal vaja valida kategooria, kus kohast talle sisu näidatakse.

Esimene valik Photo vajab kasutajalt ligipääsu kasutaja telefonis olevale pildigaleriile. Photo valiku all kuvatakse kasutajale veel kolme täpsustusvalikut. Üheks täpsustusvalikuks on Any Photo. See valik laseb rakendusel endal valida ühe suvalise pildi telefoni galeriist ning kuvab selle kasutajale. Teiseks valikuks on Pick 3, ehk on võimalik harvendada oma galeriist valitavate piltide arvu kolme peale. Kasutajale kuvatakse tema pildigalerii äpisiselt ning kasutaja valib välja kolm pilti. Olles valinud kolm pilti, ilmub võimalus vajutada Next nupule ning Rando valib välja juhusliku pildi nende kolme seast. Kolmandaks valikuks Photo valiku all on valik nimega Selfies. Nagu nimigi ütleb, on tegemist suvalise selvefoto valimisega

¹ <https://www.similarweb.com/website/9gag.com>

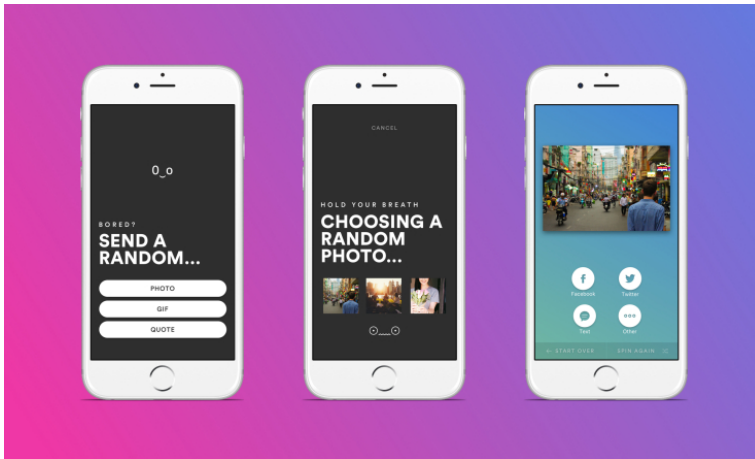
kasutaja galeriist. Loomulikult valitavate piltide eest hoolitseb iOS tarkvara, mis pildid ta tuvastab selvefotona ning mida mitte.

Teiseks valikuks on Gif, ehk liikuv pilt. Gifi alamvalikus on rohkem valikuid, kui eelmises kategoorias, kuid funktsionaalsuse poolest need pakutavad alamvalikud ei erine üksteisest. Nimelt saab siis valida teema, millist gif'i soovitakse saada ning see valik võib varieeruda.

Kolmas Quote valik sarnaneb pigem teise Gif valikuga. Kolmandas valikus järgnevad ka alamvalikud, kus jällegi tuleb valida pakutavate teemade vahel, mis võivad varieeruda. Üldjuhul on Quote alamvalikus vähem valikuid, kui Gif'i alamvalikus.

Teiseks sammuks võib lugeda kohta, kui Rando on kasutajale suvalise sisu genereerinud ning esialgu kuvatakse kasutajale udustatud pilti. Teisisõnu, kasutaja ei näe täpselt, mille rakendus on tema jaoks välja valinud. Selles punktis tuleb kasutajal teha otsus, kas soovitakse saata pilti pimesi või ikka avaldatakse pilt endale ning kasutaja näeb mis udukogu all peidus oli.

Kolmas ning viimane põhisamm rakenduses on jagamine või uuesti alustamine algusest. Rando ise pakub jagamisel kolme varianti: tekstsõnum, Twitter, Facebook. Lisaks on ka olemas Other võimalus, kuhu iOS on koondanud mõned kolmanda osapoole rakendused. Lisaks, kui peaks tekkima hoopis soov salvestada saadud sisu oma märkmerakendusse või hoopis saata kellelegi meili teel, siis see kõik on võimalik. Pimesi jagamise puhul, kõikides kanalites peale Facebooki, lisatakse pealkiri, mis mainib ära, et kasutaja saatis taolise sisuga asja pimesi, ise seda näegamata. Facebooki jagades, kasutaja lihtsalt ei näe mida ta jagab niikaua kuni on vajutatud postitamise nuppu. Pimesi jagamise puhul kuvatakse pealkiri ja pildifaili internetiaadress, kuid samas mitte pimesi jagamisel kuvatakse ikkagi kogu sisu, ehk pildifail kohe.



Joonis 1 Rando rakenduse käigud (1, 2, 3)

Üldiselt leiab Rando’st kaks sisulist asja, mida autor oma rakenduses kasutab. Nendeks on liikuvad pildid, mis kuvatakse rakenduses Giphy API kaudu ning lisaks on ka mitmeid tsitaate, mille algset allikat autor siinkohal öelda ei oska. Rando’s on kõikidel tsitaatidel lisatud alla nurka Rando rakenduse logo, mis viitaks nagu oleks tegemist rakenduse enda tsitaatide andmebaasiga. Rando on rohkem sisu jagamisele, kui sisu näitamisele keskendunud rakendus, ehk kas soovitakse selliselt kasvatada oma kasutajaskonda, et keegi jagas sõbrale sealt sisu ning ka see sõber peab kohe rakenduse alla laadima ja vaatama, mis tema esimesele sõbrale tagasi saab saata. Aja veetmiseks on selline rakendus suhteliselt aeganõudev ning kohmakas, et kiiresti uut ja põnevat sisu näha.

1.2. 9GAG – suurim meelelahutusveeb

2008. aastal loodud rakendus 9GAG on hetkel suurim meelelahutuse veebisait maailmas. Saidi Facebook’i lehel on üle 30 miljoni meeldimise. 9GAG asutaja on Ray Chan, kes ei osanud arvatagi, et sait nii populaarseks saab.

9GAG erineb suuresti eelnevalt kirjeldatud rakendusest Rando. 9GAG’i on võimalik luua kasutaja või sisse logida sotsiaalmeediate Facebook või Google+ kaudu. Kasutajana sisse logimine annab lisa funktsionaalsust rakenduse kasutamise puhul. Esiteks on kasutajal võimalik hääletada kuvatavate postituste poolt või vastu (*up-vote*, *down-vote*). See tähendab seda, et hääletades mõnda postitust ülespoole, lisandub punkt ning on suurem võimalus, et seda näeb rohkem inimesi. Alla poole hääletades kahanevad postitusel võimalused teistele

külastajatele silma jääda. Teiseks on võimalus kasutajatel osaleda postituste all käivates diskussioonidel postituse üle, ehk siis on tegu väikese kommentaarinurgaga. Kolmandaks on registreeritud kasutajal võimalik lisada ka ise postitus.

9GAG'is on külastajal suurem kontroll kuvatava sisu üle, kui Rando's. Lehe külastajal on võimalik valida esivaates kolme alamkategorია vahel: Hot, Trending ja Fresh. Hot kategoorias kuvatakse sisu, mis on hetkel kõige populaarsem. Trending kategoorias kuvatakse sisu, mis on juba mõnda aega populaarne olnud. Fresh kategoorias, nagu nimigi ütleb, kuvatakse kõige värskemad sisu, mida teised kasutajad on lisanud ning millel on veel minimaalselt häält ja kommentaare.

Rakenduse Explore vaates on võimalik kasutajal otsida endale sobivat või hetkel vajalikku sisu. 9GAG pakub ise ette üheksa valikut (Funny, GIF, Geeky, Meme, Cute Animals, Comic, Food, Timely, Design), mis võivad ühtida kasutaja otsingu soovidega.

Autor peab kolmandat kategooriat 9GAG keskkonnas vägagi suurepäraseks ning selleks on Video kategooria. Sellest kategooriast leiab videoid, mis on laetud Youtube keskkonda ning 9GAG rakenduses mängitakse neid oma videomängijast, mida on kohandatud, et kasutaja saaks kiiresti videot jagada, kui tekib soov. Video kategooriast leiab samamoodi kõike nagu ka esimese kategooria puhul. On olemas alamkategoriad, mille järgi on endale huvitava teemaga seotud sisu lihtsam otsida.

Üheks erinevuseks Rando'ga on reklaamid 9GAG'is, mille abil leht ennast üldse üleval peab. Rando, kas ei soovinud lihtsalt kasutajakogemust rikkuda reklaamidega või siis arvatavasti ei ole Rando rakendusel veel nii suuri serveri hooldustasusid, et peaks reklaamidega tulu teenima.

9GAG'il on ka välja tulnud sõnumite saatmise rakendus, esialgse nimega 9CHAT, kuid nüüdseks on sellest saanud Cookie nimeline rakendus, mis on saadaval ainult iOS ja Android seadmetele. Tegemist on suhtlusrakendusega, kus on võimalik saata ühes grupis olevatele kasutajatele kirju, kuid Cookie rakendus on saanud viimasel ajal palju halba tagasisidet.

Näiteks ei tööta rakendus korralikult, hangub. Lisaks ka ei meeldi kasutajatele uus süsteem, et olenevalt küpsiste arvust kasutaja kontrol, on tal võimalik postitada kommentaare².

9GAG sisaldab endas väga palju huvitavat, naljakat ja suvalist sisu, mis oleks autori rakenduse arendamisel väga heaks algallikaks, kuid kahjuks seda ei saa kasutada. 9GAG'i sarnasem külg autori rakendusega on naljakas sisu. Lisaks on Cookie suhtlusrakendus kaudselt sarnane autori rakendusega pakutava Chat võimalusega, kuid Cookies on siiski rohkem kasutajakesksem.

1.3. Juhuslik – meelelahutusrakenduse prototüüp

Autori rakendus juhuslik on prototüüp rakendus ning seda on esialgu lastud testida 10 kasutajal, kellelt ka võeti tagasiside rakenduse kasutamise kohta. Esimene käik, mis enne rakenduse kasutamist vaja teha, on kasutaja loomine. Rakendust kasutajata kasutada ei saa. Sisse logides avaneb eelnevatest rakendustest vähem alamkategoriaid. Üheks on Random ja teiseks Chat.

Random vaates on kasutajal võimalik näha juhuslikult valitud sisu. Sisu võib olla Gif, tsitaat, nail või mõni pilt Imgurist. Kasutaja soovi korral on võimalik vajutada "get new..." nupule ning kasutajale kuvatakse uus juhuslik sisu. Lisaks on välja toodud Facebook'i jagamise võimalus. Kuid jagatakse üldsegi kolmandat asja, kui kasutaja kuvatud nupule vajutab. Ehk teisisõnu jagamise jaoks genereeritakse uus jagamise pilt, jagamise pealkiri ja jagamise kirjeldustekst. Selles näebki autor asja võlu, et koguaeg toimub midagi uut ja huvitavat ning kahjuks ei ole kasutaja kontrollida mida tema soovib jagada. Rakendus ise ütleb, mida kasutaja soovib jagada. Loomulikult lõplik valik jääb kasutaja teha, kas ta jagab või ei jaga pakutavat sisu.

Teises vaates nimega Chat on tekitatud väike PHP jututoa liides. Ühesõnaga, kasutaja saab saata sõnumi ja ka kasutajale kuvatakse talle saadetud sõnumid. Sõnumi saatmise hetkel valitakse andmebaasist juhuslik kasutaja, kellele kirjutatud sõnum kohale läheb. Saatja ei tea, kellele tema sõnum kohale jõudis ning samamoodi ei tea ka sõnumi saaja, kellelt selline

² <http://getcookie.com/group/suggestion>

sõnum tuli. Sõnumiväljal on maksimaalne tähtede arv 126 tähemärki. Lihtsalt selleks, et vältida pikkade tekstide saatmist kasutajate vahel. Erinevalt 9GAG Cookies rakenduses ei ole võimalik Juhuslik rakenduses vahet teha, kes mis teksti on saatnud, seda muidugi kliendipoolselt.

Olemasolevatel naljasaitidel on tunduvalt palju sarnaseid elemente, kuid kuskil ei ole autor leidnud neid elemente koos eksisteerimas või siis täpselt nii juhuslikult, kui autor on soovinud.

2. Rakenduse arendus

Rakenduse arenduse peatükk käsitleb rakenduse ideed kuni rakenduse valmisprototüübini. Autor kirjeldab rakenduse esialgseid mõtteid ning mitmeid asju, mis tegelikult arendusse ei jõudnud. Lisaks kirjeldatakse rakenduses kasutatavaid süsteeme ning programmeerimiskeeli.

2.1. Esialgne rakenduse plaan

Rakenduse idee sündis juhuslikult nagu kõik siin ilmas on juhuslik. Kõige esimene idee oli vägagi ulmeline. Nimelt kasutajal oleks olnud võimalus päevas ainult ühe korra näha ja genereerida omale juhuslikku sisu. Seda enam tundus see arutelul viibinud isikutele naljakamana, kui kasutaja on oodanud pingsalt, millal on võimalus uut sisu genereerida ning siis kuvatakse talle näiteks ainult üks täht tähestikust. Mida on temal selle ühe tähega peale hakata? Jagada seda väga mõtet ei ole, sest jagamisel puudub sisu. Probleem olekski tekkinud sellest hetkest, et kui oleks rakendusse vaja uusi kasutajaid. Miks peaks keegi tegema kasutaja lehele, mida on põhimõtteliselt võimalik kord päevas kasutada ning alati ei pruugita rahul olla seal kuvatava sisuga. Kasutajal ei tule meeldegi 24 tunni möödudes, et tal ekstsisteerib konto kuskil sellisel saidil.

Kuskilt peab rakendus raha sisse tooma ning selleks käis ka arutelust läbi mitu mõtet. Esimese mõttena oleks kasutaja saanud osta endale premium konto väikese raha eest. Premium kontol ei oleks peal mingisuguseid piiranguid. Kasutaja saab juhuslikku sisu igal ajahetkel, kui selleks soovi on. Teiseks oli plaanis lubada kasutajatel lisada enda sisu rakendusse, et kõik teised võisid sattuda kellegi lisatud idee/mõttetera/pildi otsa. Sellisel juhul oleks võinud iga lisatav asi maksta väikese summa. Olgugi, et ühe lisamise pealt on summa väike, kuid kui arvestada, et mitmed kasutajad kasutavad seda teenust ning paljud ka korduvalt, siis on võimalik, et tulu teenib teine idee rohkem, kui esimene. Topelt tulu teenimiseks oleks hea lisada mõlemad variandid arendusse, kuid seda hetkel prototüübi juures ei rakendata.

Arutluselt käis ka läbi mõte, et huvitav oleks kõik lüüa segamini. Ehk kohe alguses, kui kasutaja loob parasjagu omale uut kontot rakenduse keskkonda, siis ühtegi kasutaja sisestatud infot konto loomisel ei arvestata, vaid hoopis suvaliste andmetega luuakse konto. Sisse logimisel ka tegelikult ei kontrollita kasutaja andmeid mitte millegagi, vaid antakse

kasutamiseks suvaline kasutaja andmebaasist. Tekib selline külaliskasutajate laadne olukord, kuid samas ikkagi kasutajad ei ole seotud ning iga kord võidakse kuvada teine sisu ning tekib veelgi rohkem juhuslikke olukordi rakendusse. Samas sellest mõttest loobuti, sest idee autoritele tundus see natukene liiast, et puuduvad korrektsed kasutajad.

Autor kogus esialgu mõtteid ning ideid, millist sisu võiks kuvada kasutajatele ning kuidas ja kus seda sisu saada oleks. Esimese asjana uuriti võimalusi 9GAG keskkonna integratsiooniga arendatavasse rakendusse, kuid arusaadavatel põhjustel ei soovi 9GAG oma sisu jagada teistele rakendustele. Autor ei väida, et see ei oleks võimalik, vaid eetilistel põhjustel seda ei ole tarvilik teha, sest tegu oleks vargusega.

Idee algusaegadel oli plaanis ka kasutada Instagram API't. Autor otsis välja lähedaid postitusi, mis sobiksid rakendusse, postitavaid kasutajaid (8factapp, 8dict, 8crap, factbolt, sarcasm_only) ning oli plaanis *hashtag'iga* otsida sisu Instagramist. Kahjuks selline plaan ei pidanud paika, sest tänapäeva kiiresti muutuv maailmas ei saa kunagi kindlad olla, et keegi oma API kasutustingimusi muutma ei hakka. Kuid Instagram tegi seda ning alates juunist 2016 ei ole võimalik enam *hashtag'iga* pilte välja otsida Instagramist ilma, et kasutaja oma Instagrami kontoga sisse ei logiks³. Nõnda jäigi suur allikas juhusliku sisuga kõrvale.

Järgi jäi veel Giphy API, mis on suhteliselt avatud ning lihtne implementeerida ning Imgur API, mis nõuab rohkem aega. Autoril on plaan mõlemat API't kasutada rakenduse prototüübis. Lisaks uurib autor sisu rikastamiseks juurde veel mõningaid vähem tuntumaid API'sid.

Mitmeid mõistlikke tsitaate ja nalju on autoril plaan ka rakenduse andmebaasi lisada. Naljade valimine käib juhusliku valiku teel, olenevalt sellest, kuidas nali/tsitaat autorile meeldib. Väiksemate raamatute puhul proovib autor kanda kõik tsitaadid/naljad andmebaasi, ehk tekib päris korralik kogum tsitaate ja nalju.

Tekkis küsimus ka, et millisele platvormile võiks rakenduse arendada. Kuid see küsimus sai ka kiire vastuse, sest esialgu prototüübi loomine veebirakendusena on kiirem ja lihtsam, kui native rakenduse loomine ning ka hilisemalt on võimalik veebi arendatud rakendust kiirelt üle

³ <https://www.instagram.com/developer/>

native rakenduseks üle viia Apache Cordova tarkvara abil. Autor kaalus ka Lua platvormil, mis on seotud autori igapäevase tööga, rakenduse arendamist, kuid kuna Lua platvormil on eeliseks just visuaalse poole kuvamine, siis seda ei valitud. Põhjus selles, et autoril ei ole paigas täpne disain rakenduse visuaalse poole pealt. Rakenduse keeleks on autor soovinud jätta inglise keele, sest sisu on inglise keeles rohkem ning ka vajalikkusel saab ka välismaalane aru, mis ta rakenduse lehel teha saab, kui peaks sellisele lehele juhuslikult sattuma.

2.2. Rakenduse arenduse käik

Esiialgu tutvuti üldiselt materjaliga ning mõeldi, kuidas oleks kõige parem süsteemi üles ehitada. Rakenduse arendamist alustas autor kõigepealt vajalike API-de välja otsimisega ja nende funktsionaalsuse selgeks tegemise ja testimisega. Edasi liiguti sisselogimiste ning kasutajate loomise süsteemiga ning lõpuks pandi kõik üheks tervikuks kokku. Rakenduse arenduses kasutatakse peamiselt HTML, PHP, Javascript jQuery programmeerimiskeeli. Autor, eelpool mainitult, rakenduses disaini ja kujunduse peale otseselt ei mõelnud, sest puuduvad disainimise oskused. Lihtsamad paigutused lahendati Bootstrap'iga ja osaliselt autor enda kirjutatud CSS koodiga. Põhiliselt on arendusel kasutatud PHP-d.

2.2.1. Rakenduses kasutatavad API-d

Eelpool mainituna on jäänud rakendusest välja Instagram API. Seda Instagrami kasutustingimuste muutumise tõttu. Autor oleks soovinud kasutajatele kuvada pilte Instagramist, mis on otsitud välja *hashtag*'ide kaudu. Autorile üllatuseks, alustades rakenduse arendust, oli Instagram keelanud registreeritud rakendustel, kasutajate autentimiseta, *hashtag* sisu kuvada.

Järgmiseks võttis autor ette Giphy API. Arendusprotsessis selle API lisamine mingeid probleeme ei tekitanud. API pakub rakenduse arenduse ajaks avalikku beeta võtit, mille kasutamisel saavad arendajad oma rakendust arendada. Giphy API pakub juba vaikimisi võimalust saada juhuslikku gif'i. Autor ei jäta seda võimalust kasutamata, kuid loomulikult ei saa alati jääda lootma, et mõni API teeks teiste eest töö ära. Sellepärast kasutab autor rakenduses võimalikult palju erinevaid endpoint'e (vt koodinäide 1).

http://api.giphy.com/v1/gifs/random?api_key=API_KEY&tag=TAG

Koodinäide 1 Üks URL'idest, kuhu tuleb päring teha. API_KEY ja TAG asemel tuleb sisestada õiged andmed.

Lisaks random endpoint'ile kasutatakse translate ja search endpoint'e. Translate endpoint tähendab seda, et põhimõtteliselt otsitakse kõiki gif'e, kuhu on lisatud tähendusega teksti. Translate endpoint'i kasutavad hetkel teadaolevalt sellised rakendused nagu Slack, Hipchat, Wire ja Dasher (Giphy API Documentation 2016). Search endpoint annab kõige suurema valiku vasteid. Olenevalt otsitavast fraasist võib vastete arv ulatuda kuni 34 430 liikuva pildini. Eelpool mainitud arvu sai autor otsides *funny* nimelisi gif'e. Search endpointis on võimalik kasutada limit ja offset parameetreid, mille abil leiab autor juhusliku pildi. Iga tag'i kohta uurib autor välja, palju on pilte nende taga ning siis genereeritakse juhuslik arv vahemikus 0 kuni eelpool nimetatud kindla arvuni. Kõigis kolmes endpoint'is kasutatakse ühtesid ja samu fraase: mystery, quote, random, minions, hilarious, coding, funny, office, comedy, butterflies, savage, wtf, fail, trending, dumb, kevin hart, breaking bad. Neid on võimalik igal ajahetkel juurde lisada ning ära võtta vastavalt autori soovile. Lisaks soovib Giphy, et nendelt saadud sisu juurde, tuleks lisada väike Giphy logo (vt joonis 2). Neid on mitmes variandis, kuid autor valis kõige sobilikuma enda rakenduse jaoks.

POWERED BY **GIPHY**

Joonis 2 Giphy API logo, mida kasutatakse arendatavas rakenduses.

Selleks, et kasutada rakendust laiemas avalikkuses, tuleb registreerida oma rakendus ning siis ka see Giphy'sse ülevaatusse anda, kus kõigi asjade sujumisel antakse iga rakenduse enda API võti, millega võib rakenduse avalikuks teha. Tuleb kirjutada, kuidas rakendus kasutab Giphy API't ning veel paarile küsimusele vastata. Kindluse mõttes tegi ka autor läbi selle kadalipu, et võimalikke ebameeldivusi vältida testperioodi ajal. Autor mainis täidetavas vormis, et rakendust testivad maksimaalselt 20 inimest ning tegemist on autori lõputööga. Üllatuseks saadi vastus järgmisel päeval Giphy'lt. Vastuses sooviti edu autorile tema koolitööks ning ka samas lubati avalikku beeta võtit kasutada, et taolise testperioodi jaoks ei ole autoril isiklikku võtit vaja. Probleemide korral soovitati julgelt kirjutada ning autorit üllataski selline personaalne lähenemine. Lisades on välja toodud näited, kuidas Giphy'st andmed kätte saadakse.

Imgur API lisati kõige viimasena rakendusse, sest see tundus autorile kõige keerukam. Kuid täpsemalt uurides dokumentatsiooni selgus, et Juhuslik rakenduse toiminguteks ei ole vajagi kasutajaid autentida, mida algul eeldati, et on vaja. Imgur API kasutama hakkamine väga ei erine Giphy omast. Tuleb luua uus rakendus ning siis on võimalik kasutada Client ID-d rakendusest sisu kätte saamiseks. Client ID lisatakse GET päringu päisesse, mille kaudu rakendust identifitseeritakse ja õnnestumisel tagastatakse JSON kujul andmed. Võimalus on ka veel JSONP ja XML kujul andmeid tagastada, kuid autor kasutab hetke olukorras JSON lahendust. Imgur'is kasutatakse topic nimelist endpoint'i. Seda sellepärast, et kõige rohkem on võimalik sisu saada just sellest endpoint'ist, ilma et peaks autoriseerima kasutajat Imgur'iga. Imgur jaoks kasutab autor rakenduses cURL'i (vt koodinäide 2). Kõigepealt valitakse välja juhuslik teema 11 topic'u seast. Edasi tehakse päring Imgur API'le ning tagastatud andmete seast valitakse välja taaskord suvaline link pildifailile. Mõningatel juhtudel võib sattuda ka link, mis ei ole otseselt pildifail. Sellisel puhul võetakse selle lingi ID ning tehakse uus päring Imgur'isse sooviga saada andmeid pakutud ID kohta ning sealt edasi juba saadaks õige pildifaili link.

```
$ch = curl_init("https://api.imgur.com/3/topics/". $topic);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_HTTPHEADER, array(
    'Authorization: Client-ID {CLIENT_ID}'
));
$img = json_decode(curl_exec($ch));
curl_close($ch);
```

Koodinäide 2 PHP cURL päring Imgur API'le. Topic muutuja saadakse juhuslikuna topic massiivist, kus on eelnevalt paika pandud, millised teemad olemas on. CLIENT_ID on vajalik rakenduse autoriseerimiseks ning img muutujas on päringule vastuseks saadetud andmed.

2.2.2. Rakenduses kasutatav muu sisu

Lisaks ka korraliku dokumentatsiooniga API'dele, kasutatakse rakenduses ka kohalikust andmebaasist saadavat materjali ning ka lisaks muid leitud allikaid, kus on võimalik süsteemselt infot kätte saada. Rakenduse andmebaasi on autor lisanud tsitaate, nii internetist leitud, kuid ka raamatutest otsitud. Omast andmebaasist andmete pärimisega on eelis just

eelkõige rakenduse toimimise kiiruses. Kohati võib teistesse rakendustesse päringu tegemine võtta mitmeid sekundeid. Lisaks on kasutatud ka kolme aadressi, kuhu päringuid tehakse. Tagastatud andmetes õige lingi kätte saamiseks on autor kasutanud Regular Express'i. Kõige harvemini kasutatakse autori kirjutatud juhuslikke lauseid, kus absoluutselt puudub loogika. Neid lauseid kuvatakse umbkaudu 10% korral ja väiksemana, kui teist sisu. Autor soovis lihtsalt tuua rohkem segadust ja ka üllatusmomenti rakendusse.

2.2.3. Rakenduse funktsionaalsuse arendus

Kõige esimene kasutatav funktsioon oli kasutaja loomise võimalus. Kasutajal on vaja sisestada enda õige meiliaadress ning oma parool kaks korda, teine kord on parooli kinnitamiseks. Uuele kasutajale saadetakse meili kinnitusvorm. Seda tehakse selleks, et vältida meeletutes kogustes pahatahtlike kasutajate kontode tegemist. Kasutaja saab alles pärast kinnituslingi avamist oma kontosse sisse logida. Kasutajal on võimalus ka uut parooli küsida, juhul kui hetke parool on ununenud. Kasutajale saadetakse registreeritud meilile parooli muutmise unikaalne link, kus tuleb sisestada lihtsalt uus parool.

Juhuslike numbrite genereerimisel rakenduses kasutatakse *mt_rand* PHP funktsiooni, mis kasutab Mersenne Twister algoritmi. Lisaks veel *mt_rand* on neli korda kiirem kui tavaline *rand* funktsioon ning samuti genereerib vähem ettearvatama juhusliku numbri.

Kasutajate paroolide krüpteeritakse PHP *password_hash* funktsiooniga. PHP Password Hashing API on PHP poolt välja arendatud paroolide loomise ja haldamise funktsionaalsus, et lihtsal ja kiirel viisil kontrollida kasutaja sisestatud parooli andmebaasis oleva räsiväärtusega. Rakenduses kasutatakse kahte järgmist funktsiooni:

- *Password_hash* – genereerib parooliräsi.
- *Password_verify* – kontrollib, kas sisestatud parool on võrdväärne andmebaasist võetud parooliga.

Pärast sisse logimist on kasutajal võimalik valida üks kolmest valikust – Chat, Random, About. Chat valiku all on võimalus saata sõnum juhuslikule kasutajale. Seal vaates on kasutatud ka jQuery abi. Täpsemalt selle jaoks, et uuendada iga kolme sekundi tagant sõnumite näitamise välja. Vältimaks suuremat ajakadu sõnumi saatmise ja sõnumi kätte

toimetamise vahel. Edasi liikudes Random valiku alla on võimalik näha juhuslikult valitud pilti, tsitaati või nalja. Lisaks on seal võimalus ka saadud sisu jagada Facebook'is.

Facebook'is jagamiseks tuleb samuti luua rakendus <https://developer.facebook.com> lehel ning pärast seda on võimalik kasutada Facebook'i pakutavaid teenuseid. Autor kasutab rakenduses ainult Javascript SDK'd. Selleks, et ka Facebook rakendus avalik ja saadaval oleks kõigile kasutajate, tuleb täita mõningad välja. Näiteks tuleks lisada oma rakenduse logo ja domeeni aadress, kus kohast lubatakse päringuid teostada. Lehele, kus on soovi rakendust kasutada, tuleb lisada script, mis käivitab Facebook'i SDK (vt koodinäide 2).

```
<script>
    window.fbAsyncInit=function(){FB.init({
        appId:'your-app-id',
        xfbml:true,
        version:'v2.5'
    });};(function(d,s,id){var js,fjs =
d.getElementsByTagName(s)[0];if(d.getElementById(id)){ret
urn;}js = d.createElement(s); js.id = id;js.src =
"//connect.facebook.net/en_US/sdk.js";fjs.parentNode.inse
rtBefore(js, fjs);}(document, 'script', 'facebook-
jssdk'));
```

```
</script>
```

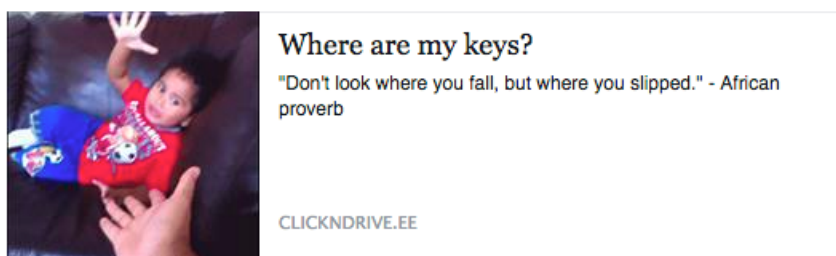
Koodinäide 3 Script, mis on vajalik lisada kohe pärast html body elemendi algust. Script käivitab Facebook SDK.

Jagamise jaoks kasutab autor Facebook'i pakutavat *Feed Dialog* võimalust. See siis tähendab seda, et kasutajal on võimalik postitada individuaalseid postitusi. Samas on kasutajal ka võimalus lisada enda mõtteid eelgenereeritud sisule Juhuslik rakenduse poolt. Rakenduses kasutatakse jagatava sisu loomiseks samuti juhuslikku generaatorit. Ehk kasutaja ei jaga seda, mida talle Random lehel näidatakse, vaid jagamiseks genereeritakse omakorda uus sisu (vt joonis 4). Jagamisel antakse kaasa neli parameetrit:

- *Method* – millist meetodit soovitakse kasutada. Hetkel on selleks *feed* meetod.
- *Link* – link, kuhu suunatakse kasutaja, kes on Facebookis kuvatud postitusele klikanud.
- *Picture* – kuvatav pilt jagatud postituse juures.

- *Name* – postituse pealkiri.
- *Description* – postituse kirjeldus.

Say something about this...



🔒 Only Me ▾

Cancel

Post to Facebook

Joonis 3 Kasutajale kuvatav jagamise aken. “Say something about this...” reale on võimalik kasutajal kirjutada midagi enda poolt. Lisaks on olemas pilt, pealkiri ning kirjeldus, mis on juhuslikult genereeritud.

Facebook’i jagamisega lõpeb ka Random lehel funktsionaalsus.

Rakenduse tutvustamiseks on olemas järgmiseks About leht. About lehel on kirjutad, mida rakendus endas sisaldab ning millised on rakenduse kasutamise võimalused. Lisaks saab ka About lehel anda ideid, millist sisu võiks rakenduses lisaks kuvada. Loomulikult on samas oodatud igasugune tagasiside rakenduse kohta. Allpool on kuvatud tavalised Facebook’i jagamise ja like nupud, kui kellelgi peaks soovi olema rakendust like’da.

2.2.4. Rakenduse andmebaas

Rakenduses kasutatakse MySQL andmebaasi ning andmebaaside loomisel kasutati phpMyAdmin andmebaasi liidest. Kõik andmebaasis talletatavad andmed kodeeritakse UTF-8 meetodi järgi. Andmetabeleid on baasis kokku neli:

- Kasutajate tabel – tabel, milles hoitakse kõigi kasutajate andmeid.
- Sõnumite tabel – tabel, milles hoitakse kõiki saadetud sõnumeid kasutajate vahel.
- Tsitaatide tabel – tabel, kuhu on kogutud raamatutest ja veebist tsitaadid.
- Naljade tabel – tabel, kus hoitakse erinevaid nalju

Andmebaasiga ühendumiseks ja päringute tegemiseks kasutakse mysql-i laiendit. On mitmeid eeliseid, miks otsustati just seda kasutada. Esiteks on seda soovitatud kasutada juhul kui on installitud vähemalt 4.1.3 MySQL versioon. Autori rakendus kasutab 5.6 versiooni. Teiseks on objekt-orienteeritud kasutajaliides, mida on mugavam kasutada ning lisaks on ka mainitud veel ettevalmistatud andmete toetus, mitmik andmete toetus, tehingute toetus, tõhusad silumise võimalused ning seotud serveri tugi.

Kasutajate tabelis hoitakse kasutajate andmeid ning see tabel on lahtrite arvult kõige suurem rakenduses (vt tabel 1).

Id	Int(11) AUTO_INCREMENT	Kasutaja järjekorra number tabelis, muud otstarvet sellel otseselt ei ole.
Email	Varchar(255)	Kasutajate meiliaadresside hoidmiseks
Password	Varchar(255)	Kasutajate parool krüpteeritud kujul
User ID	Varchar(32)	Kasutajate unikaalne id, mille kaudu ka on loodud seosed teiste tabelitega
User verified	Varchar(256)	Määramaks ära, kas kasutaja on oma konto identifitseerinud. Kui on siis kustutatakse identifitseerimisräsi ära ja pannakse asemele 1. Mis tähendab, et kasutaja on oma konto identifitseerinud.
New password	Varchar(64)	Tavaliselt on see lahter tühi. Kuid kui kasutaja on soovinud uut parooli, siis genereeritakse siia 64 kohaline räsiväärtus. Mis

		taaskord kustutakse, kui kasutaja on oma parooli muutnud.
Last login	Timestamp CURRENT_TIMESTAMP	Hoitakse kasutaja viimast sisselogimist
Created date	Timestamp CURRENT_TIMESTAMP	Kasutaja loomise kuupäev
Updated date	Timestamp CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP	Näitab, mis ajal on kasutaja andmeid andmetabelis viimati muudetud.

Tabel 1 Kasutajate tabeli ülevaade.

Sõnumite tabelis on vähem lahtreid, kuid tabeli pikkus võib olla mitmetes kordades suurem, kui kasutajate tabel. Siin tabelis hoitakse kõiki saadetud sõnumeid (vt tabel 2).

Id	Int(11) AUTO_INCREMENT	Sõnumi järjekorra number.
From user	Varchar(32)	32 kohaline räsiväärtus, mis kujutab endast saatja unikaalset id-d.
User to	Varchar(32)	32 kohaline räsi, mis kujutab endast sõnumi saaja unikaalset id-d.
Message	Varchar(126)	Sõnum, mis saadeti. Piiratud pikkus on 126 tähemärki.
Created date	Timestamp CURRENT_TIMESTAMP	Sõnumi saatmise kuupäev.

Tabel 2 Sõnumite tabeli ülevaade.

Tsitaatide tabelis hoitakse tsitaate, mis on sisestatud sinna autori poolt (vt tabel 3).

Id	Int(11) AUTO_INCREMENT	Tsitaadi järjekorranumber.
----	---------------------------	----------------------------

Quote	Varchar(255)	Tsitaat, mis siis tabeli järgi saab maksimaalselt olla 255 tähemärki pikk.
Author	Varchar(255)	Enamustel tsitaatidel on ka autor.
Source	Varchar(255)	Salvestatakse tsitaadi allikas, juhuks kui seda peaks vaja minema.
Created date	Timestamp CURRENT_TIMESTAMP	Tsitaadi lisamise kuupäev

Tabel 3 Tsitaatide tabeli ülevaade

Naljade tabelis hoitakse nalju, mis võivad olla nii lühikesed, kui ka pikad. Naljade tabelis on kõige vähem lahtreid (vt tabel 4).

Id	Int(11) AUTO_INCREMENT	Nalja järjekorranumber.
Content	Text	Nalja või fakti pikkus saab olla maksimaalselt 65 535 tähemärki pikk.
Source	Varchar(255)	Allikas, kus andmed on saadud, juhuks kui seda on vaja.
Created date	Timestamp CURRENT_TIMESTAMP	Lisamise kuupäev.

Tabel 4 Naljade tabeli ülevaade

2.2.5. Rakenduse struktuur

Rakenduse struktuur on püütud hästi lihtsana hoida ning rakendust arendades on autor püüdnud nii arendada, et veebilehitseja aadressiriba oleks võimalikult puhas ja ei kuvataks rakenduse poolt imelikke aadresse. Järgnevalt on välja toodud näited imelikust aadressist (vt koodinäide 4) versus puhtast veebiaadressist (vt koodinäide 5). Lisaks on puhtamate

aadresside usaldusväärsus suurem, inimestele jäävad need paremini meelde ning ka otsingumootorid leiavad neid aadresse paremini (How to make clean URLs, kuupäev puudub).

`http://www.desiquintans.com/index.php?page=articles`

Koodinäide 4 Segadust tekitav aadress.

`http://www.desiquintans.com/articles/`

Koodinäide 5 Puhas ja ka ilusam aadress

Rakenduses olevad kategooriad on sorteeritud eraldi kaustadesse. Peamised abifailid asuvad koos assets kausta ja kategooriate kaustadega ühel tasandil (vt joonis 4).

```
| juhuslik -
    | about -
    | assets -
        | css -
            | bootstrap.min.css
            | main.css
        | fonts -
            | CoolveticaRg-Regular.eot
            | CoolveticaRg-Regular.ttf
            | CoolveticaRg-Regular.eot
        | imgs -
            | logo.png
            | PoweredBy_200px-White_HorizText.png
        | js -
            | main.js
    | chat -
        | index.php
    | forgot -
        | index.php
    | mail -
        | newpw.html
        | verify.html
    | random -
        | index.php
    | register -
        | index.php
    | cooldata.php
    | forgotpw.php
    | ftr.php
    | getmessages.php
    | hdr.php
    | ihaveanidea.php
    | index.php
    | ini.php
    | login.php
    | random.php
    | register.php
    | sendmessage.php
    | setnewpw.php
    | verify.php
```

Joonis 4 Juhuslik rakenduse failide struktuur.

Lisaks veel assets kaustale on näha mail kausta. Seal hoitakse meilide saatmiseks vajalikku näidis HTML faili. Kus asendatakse teatud tekst õige tekstiga ja saadetakse kasutajale. About, chat, random, register ja forgot kaustades on näha ainult index.php faili. Sellega saavutabki autor selle, et aadressid oleksid võimalikult puhtad. Igas index.php failis vajatakse hdr.php faili, mis sisaldab osa, mis on igas index failis samasugune. Hdr.php faili omakorda veel lisatakse ini.php fail, see on fail sisaldab andmeid andmebaasi ühendamiseks ning ka alustab PHP sessiooni. Ftr.php failiga on sama lugu nagu hdr.php failiga. Seda faili kasutatakse igas index failis ning ka seal sisaldub informatsioon, mis on igalpool sama (vt koodinäide 6).

```
<?php
    if (isset($_SESSION["userid"])){
        echo '<a href="?logout">Log out</a>';
    }
?>
<br>
</div>
<script type="text/javascript"
src="../../juhuslik/assets/js/main.js"></script>
</body>
</html>
```

Koodinäide 6 ftr.php faili sisu.

Cooldata.php failis hoitakse random.php faili jaoks vajalikke massiive. Forgotpw.php, login.php, verify.php failid on tavalised abifailid, millega kontrollitakse kasutajate andmeid ja ka muudetakse neid. Getmessages.php ja sendmessages.php on chat osa jaoks vajalikud failid, vastavalt siis sõnumi saatmiseks ja sõnumite vastuvõtmiseks.

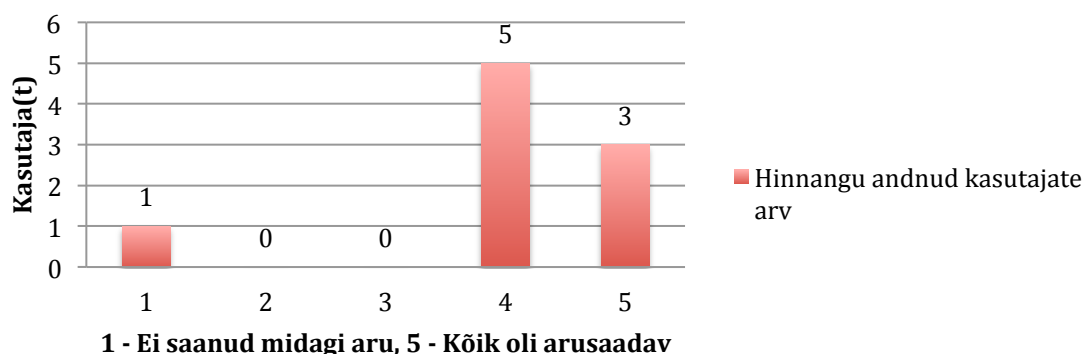
3. Testkasutajate tagasiside ning veebilehel käitumise analüüs

Rakenduse funktsionaalsuse testimises osales kokku 10 isikut. Testijateks olid noored inimesed, vanuses 19 kuni 25. Üks testijatest oli naissoost, teised üheksa isikut olid meessoost. Rakenduse testperiood kestis kokku viis päeva. Rakenduse autor, andes testlingi testijatele üle, soovis, et testijad prooviksid kasutada rakendust mitmel päeval, kuid tegelikkuses kasutas rakendust vaid üks testija (10% testkasutajatest), konto loomisele järgneval päeval. Tagasiside küsitlus viidi läbi Google Forms platvormil ning näidisküsitlus on lisatud lisadesse.

3.1. Tagasiside analüüs

Esimene tagasiside, mis tegelikult saadi igalt testijalt oli väide, et kinnituskirja ei ole nende postkasti tulnud. Autor suunas kiirelt kasutajad nende endi rämpsposti kausta, kus link kasutajakonto kinnitamiseks ka täitsa olemas oli. Lisaks viis autor ka veebilehel uuenduse sisse seoses kasutajate informeerimises, et kontrollida kindlasti ka spämmikausta. Mõningate kasutajate loomise puhul ei kasutatud pikemaid paroole, kui 8 tähemärki. Backend'is kontrollitakse, et kasutaja parool oleks vähemalt 8 märki pikk, kuid registreerimisvormi juures seda informatsiooni kasutajale ei kuvatud. Eelnenud täpsustused lisas autor kiiremas korras rakendusse. Mõningatel juhtudel ei saadud üldsegi aru, mida on võimalik rakenduses korda saata. Seda punkti kajastati ka tagasiside küsitlusel (vt Tabel 1). Kasutajad ei saanud esmalt lehele jõudes aru, et kuhu tuleb edasi liikuda ning tuli jagada näpunäiteid. Autor lisas testperioodi kolmandal päeval esialgsed vihjed, mida on võimalik hetkel külastataval lehel teha.

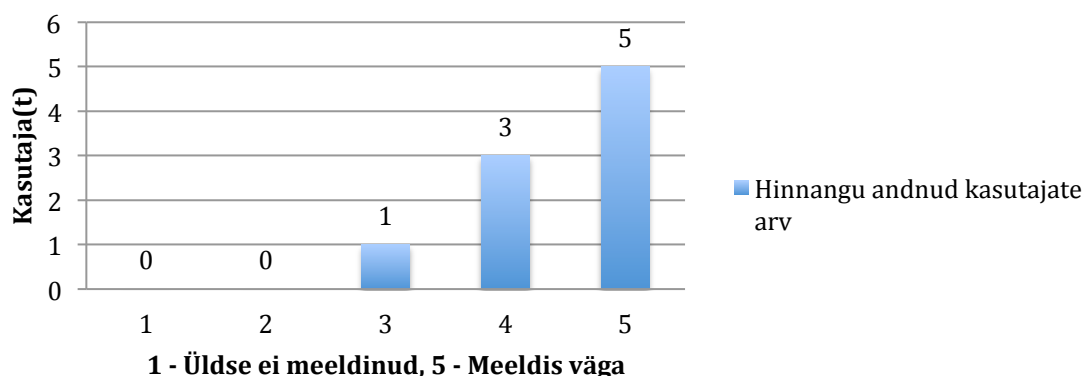
Kasutajate arusaamine lehel toimuvast, alates hetkest millal lehe saabuti



Tabel 5 Testkasutajate hinnang prototüübist arusaamise kohta esimesel külastusel.

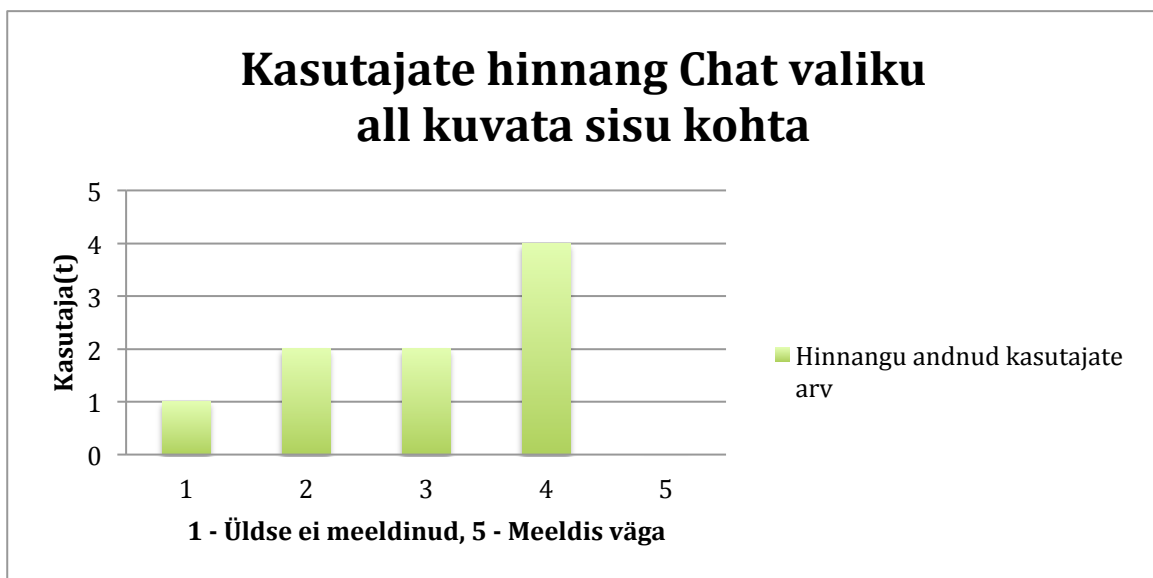
Tagasiside põhjal võib väita, et kasutajatele enamasti meeldis Random kategooria all kuvatav sisu (vt Tabel 2). Kuid oli ka märkusi mõningate asjade kohta. Näiteks üks kasutaja ei saanud aru, mille pärast kuvati lihtsalt mõnda tähte või numbrit. Kuid rakenduse mõte ongi selles, et kohati näidata täiesti seosetut ja mõttetut sisu, millest osad inimesed aru ei pruugigi saada. Sai *About* lehel on võimalik saata ka oma ideid rakenduse arendajale ning üks kasutaja soovis, et rakenduses võiks olla rohkem liblikaid. Rakenduse autor kaalub seda pakkumist ning võib-olla tõesti tulevikus leiab rakendusest rohkem liblikaid.

Kasutajate hinnang Random valiku all kuvata sisu kohta



Tabel 6 Kasutajate hinnang rakenduse Random valiku all kuvatavale sisule.

Autoril olid suured ootused Chat kategooria puhul. Tundus, et see on tegelikult uudne asi ning võiks kasutajatele meeldida, kuid tagasisides antud hinnete põhjal osutus asi vastupidiseks (vt Tabel 3). Nagu tabeli põhjal on näha, siis 55,6% kasutajale meeldis keskmiselt või alla keskmise *Chat* lehe sisu. Põhjuseks, et testperioodi alguses ei olnud lisatud juhendit, kuidas *Chat* lehel toimida, siis jäi ühele kasutajale arusaamatuks, miks ta ei näe sõnumite saatja nime. Lisaks soovitati ka lisada vähemalt enda kirjutatud tekstid, et oleks kasutajal endal näha, mida ta on kirja pannud. Seda mõtet autor kaalus ning esialgu jõudis järeldusele, et ei soovi seda funktsionaalsust lisada, sest kasutajal ei ole mingisugust vajadust enda saadetud sõnumeid näha. Tuues võrdluseks pärismaailma, siis keegi ei mäleta täpselt, mida ta kellelegi ütles. Siinkohal võib keegi väita, et inimene ei mäleta ka täpselt, mida keegi talle ütles, et miks siis vanu vestlusi kuvatakse *Chat* vaates. Põhjus selles, et muidu ei tekiks mängulisust ning tekiks küsimus, kaua peaks kasutajale näitama saadetud sõnumit. Samuti oleks siis otstarbekas vahendada sõnumeid ainult online kasutajate vahel.



Tabel 7 Kasutajate hinnang rakenduse Chat valiku all kuvatavale sisule.

Üldiselt võib väita, et kasutajatele meeldis enim *Random* osa prototüübist. Koguni neli kasutajat on tagaside viimases vabas tekstivormi lahtris ülistavalt kiitnud *Random* osa. Ühele kasutajale jäis silma väga algne disain ning soovis, et rakendus võiks näha visuaalselt ilusam välja. Autor on selle väitega täielikult nõus, kuid arendatava prototüübi loomisel ei pööratud tähelepanu niivõrd rakenduse disainile, kuivõrd hoopis funktsionaalsuse demonstreerimisele.

3.2. Testkasutaja veebilehel käitumise analüüs

Kasutajate tegevusi salvestati serveris olevasse tavalisse *.txt* faili. Esmalt oli autoril plaan kasutada näiteks Google Analytics teenust, kuid kuna selgus, et Google Analytics teenus pole kõigile niisama lihtsalt kättesaadav, siis võeti kasutusele kõige lihtsam variant testgrupi jaoks.

Andmebaasist ning ka kasutaja käitumise logidest on võimalik näha, et üks kasutaja on proovinud kõige lihtsamate vahenditega rakenduse toimimist takistada või vigu leida. Eelmainitud kasutaja proovis peamiselt SQL Injection'i kaudu asju lõhkuda (vt koodirida 1). Esialgu oli näha ning ka kasutaja ise väitis, et ei suutnud midagi ära lõhkuda rakenduse töös. Kuigi autor testkasutajatelt taolist testimist ei nõudnud, siis oli selle kasutaja testimisel lisandväärtus täitsa olemas ning rakenduse autor on tänulik sellele testkasutajale.

```
``; DROP TABLE USERS;
```

Koodinäide 7 Andmebaasist välja võetud SQL Injection lause, mida testkasutaja on proovinud.

Kasutajad logisid rakendusse sisse keskmiselt kaks korda. Kõik kasutajaid sisenesid vähemalt ühe korra saidile ning maksimaalselt logiti sisse neli korda. Autor ei oska välja tuua põhjust, miks üks kasutaja on niipalju kordi sisse loginud 10 minutiliste vahedega. Kuid võib oletada, et kasutaja proovis süsteemi töökindlust. Teisel kasutaja, kes on neli korda sisse loginud, on logimiste vahel vähemalt neli tundi vahet. Tundub, et kasutaja külastas saiti nii nagu võiks keskmine kasutaja külastada. Ehk hetkedel, millal inimestel on igav ja nad otsivad aja mööda saatmise võimalust. Pooled kasutajad logisid rakendusse vaid korra sisse.

Vestlusest said põhimõtteliselt osa kõik kasutajad. Kui kasutaja ise ei saanud sõnumit, siis kellegi sõnum jõudis temani ikkagi ning kui mõni kasutaja ei saanud mitte ühtegi sõnumit, siis logidest võime välja lugeda, et tema oli usin sõnumite saatja. Keskmiselt saadeti sõnumeid ühe kasutaja kohta 4,1 korda. Kokku saadeti 41 sõnumit. Üks kasutaja ei saanud mitte ühtegi sõnumit, kuid temale saadeti kokku 6 sõnumit. Kasutaja, kes proovis süsteemi lammutada, saatis kokku 19 sõnumit, kuid ta ei saanud ise mitte ühtegi sõnumit. Süsteemis on ka kolm kaduma läinud sõnumit. Otseselt ei ole need sõnumid kadunud, vaid need on saanud kasutajad, kes ei ole oma kontot meili kaudud kinnitanud. Autor parandas vea alles pärast

testperioodi lõppu, et sõnumeid saadetak ainult kasutajatele, kes on oma konto õigsust kinnitanud.

Random kategoorias sooviti juhuslikku sisu kokku 353 korda, keskmiselt ühe kasutaja kohta 34,5 korda. Kokku kõige rohkem päringuid tehti ühe kasutaja kohta 127 päringut ning nendest 94 päringut tehti ühe sessiooni vältel. Kõige vähem tehtud päringute arv kasutaja kohta on 7 ning need kõik tehti ühe sessiooni ajal. Siit võib järeldada, et osadele kasutajatele ikkagi meeldib suvaline ja meelelahutuslik sisu. 19,8% kordadest näidati kasutajatele Giphy API'st random gif'i. 16,7% kordadest näidati serveri andmebaasi talletatud tsitaate. 18,1% puhul kuvati kasutajatele Imgur API kaudu saadud pilti. 17,8% kordadest nähti filmidest välja võetud tsitaate. 13,5% näidati juhuslikke nalju või fakte ning suvalise tähe või numbri otsa satuti 14,1% juhtudest (vt tabel 4).

Osakaal	Kirjeldus
19,8 %	Giphy sisu
18,1 %	Imgur sisu
17,8 %	Filmide tsitaadid
16,7 %	Tsitaadid andmebaasist
14,1 %	Suvaline täht/arv
13,5 %	Naljad/faktid andmebaasist

Tabel 8 Edetabel kõige enam kuvatud sisust

Autor jääb üldiselt rahule sellega, et iga kasutaja soovis Random valiku all sisu mitmeid kordi uuendada, ehk juhuslikult kuvatud sisu intrigeeris kasutajaid. Rahule ei jäetud sellega, et kasutajaid ei köitnud Chat valiku all sisalduv funktsionaalsus. Autor arvab, et selles on oma jume siis, kui kasutajaid on mitmeid kordi rohkem ja nad ka kirjutavad mitmeid sõnumeid. Samas oleks ka võimalik luua algoritm, mis koordineerib seda, et suurem tõenäosus on saada sõnumit kasutajal, kes on ise ka rohkelt sõnumeid saatnud. Autor lootis, et rohkem kasutajaid logib sisse rakendusse ka järgnevatel päevadel, kuid kahjuks see ei pidanud paika.

Kokkuvõte

Töö eesmärk prototüübi arenduse poolt sai autoril täidetud ja on esialgse tulemusega rahul. Rakenduse prototüübile on arendatud ootuspärane funktsionaalsus ning peaks andma aimu, mida üldisemas pildis arendada soovitakse.

Testgrupi testimise järelendusena võib autor väita, et osa testitavatest kasutas rakendust päris usinalt. Seda siis Random sisu kuvamise poolt. Pettumust valmistas kasutajate vähene aktiivsus järgnevatel päevadel rakenduses. See põhimõtteliselt puudus. Edasiseks arenduseks võiks ka välja selgitada, kas määras asja siinkohal testgrupis osalenute väike arv või on tegemist siiski liialt igava rakendusega. Autor ei saa ka mainimata jätta, et päris suure aja kulutas autor ise juhusliku sisu kuvamisele, alati lootuses, et järgmine asi on lähedam ja naljakam. Samamoodi ka otsides lisatavaid materjale jäädi mõnikord liialt pikaks ajaks nalju, tsitaate lugema ja pilte vaatama.

Rakenduse edasiarenduse koha pealt on soov autoril rakendus ka lõplikult välja arendada ja korralik disain lasta luua. Lisaks pärast rakenduse lõppversiooni valmimist võiks olla plaan arendada rakendus ka nutitelefonidele Apache Cordova'ga näiteks.

Rakendusse on võimalik lisada alati sisu juurde ning ka põhimõtteliselt võib sinna lisada kõike, sest tegemist on juhusliku rakendusega. Pärast testgrupiga testimist tuli autorile palju mõtteid juurde, mida rakendusse lisada. Näiteks on võimalik, et tulevikus lisatakse ka Google Maps API, mis näitab juhuslikku Streetview pilti.

Kasutatud kirjandus

Henley, D. (2012). *Inspirational Quotes Succes, Motivation, Effort, Adversity, & Mindset*. Henley Sports Performance

El-Manzalawy, W. (2010) *150 Quotes About Success And Life*. Smashwords Edition.

Goldstein, J. (2014). *101 Amazing Facts*. Andrews UK Limited.

Parker, J. (2012). *50 Best Jokes Ever*. Smashwords Edition.

Summersdale Publishers Ltd. (2010). *NOW PANIC AND FREAK OUT*. Chichester: Summersdale Publishers Ltd.

Perez, S. (2016, 4. mai). Rando is a random social sharing app that lets you play Russian Roulette with photos, GIFs and more. Loetud aadressil <http://techcrunch.com/2016/03/04/rando-is-a-random-social-sharing-app-that-lets-you-pay-russian-roulette-with-photos-gifs-and-more/>

Ren, F. (2012, 24. juuli). How Ray Chan started 9GAG, and a career in fun. Loetud aadressil <http://www.meldmagazine.com.au/2012/07/9-gag-ray-chan/>

Gannes, L. (2012, 12. aprill). Meet 9GAG, the Community Comedy Site That's Growing Like Crazy. Loetud aadressil <http://allthingsd.com/20120412/meet-9gag-the-community-comedy-site-thats-growing-like-crazy/>

Quintans, D. (kuupäev puudub). How to make clean URLs. Loetud aadressil <http://www.desiquintans.com/cleanurls/>

Ingram, M. (2012, 23. juuni). Is modern technology creating a culture of distraction?. Loetud aadressil <https://gigaom.com/2012/06/23/is-modern-technology-creating-a-culture-of-distraction/>

Github.com (kuupäev puudub). Giphy API Documentation. Loetud aadressil <https://github.com/Giphy/GiphyAPI>

W3Schools.com. (kuupäev puudub). PHP mt_rand() Function. Loetud aadressil http://www.w3schools.com/php/func_math_mt_rand.asp

Imgur. (kuupäev puudub). Imgur API Version 3. Loetud aadressil <https://api.imgur.com/>

Facebook. (kuupäev puudub). Facebook SDK for JavaScript. Loetud aadressil <https://developers.facebook.com/docs/javascript>

Summary

Title: Random - Entertaining Application Prototype

The goal of this Bachelors thesis was to develop a functional prototype of random entertaining application. This goal was reached and author is satisfied with development result. Users can see and feel what was the idea about.

After the test period, author can say that most of the users used Random content repeatedly. Author was disappointed in lack of users activity after they had registered and used application for the first time. For further reasearch should find out what was the reason for this kind of inactivity from users, was there too few people in the test group or is this application truly boring. Author mentions that the search for funny and interesting content took too much time sometimes. He was just like distracted from development by cool and funny content.

Author will continue developing this application and someone is needed to design this application also. After first official version there will be a development for smartdevices with Apache Cordova.

There is always possible to add some new content to this application. There can be added anything no matter what, because it is random application. After this application was tested with test group, author came up with the new idea, that there should be also Google Maps API Streetview included from random place on earth.

LISAD

Giphy'st andmete kätte saamine random endpoint järgi:

```
if ($fr == 0){ // $fr on viide juhuslikule arvule
    $tag = "&tag=".$tagarray[mt_rand(0,count($tagarray)-1)]; // võtame
    juhusliku tag elemendi massiivist, kus asuvad fraasid
    $content =
    json_decode(file_get_contents("http://api.giphy.com/v1/gifs/random?ap
    i_key=dc6zaTOxFJmzC".$tag)); // tõmbame JSON sisu, mis on saadaval
    genereeritud url-il ja kohe ka dekodeerime selle.
    $randomimg_url = $content->data->image_original_url; // liigume
    objektipuus pildi originaal aadressi juurde ja omistame selle
    muutujasse
    echo '<br>'; // kuvame veebilehel
}
```

Javascript funktsioon sisu jagamiseks Facebook'i

```
function postToFeed(){ // funktsioon Facebook'i jagamiseks
    var data = fb_share_data; // kirjeldame kohaliku muutuja, kuhu
    määrame eelnevalt genereeritud objekti.
    var obj={ // kirjeldame objekti, mis sisaldab postitavaid andmeid
        method: 'feed', // meetod, kuidas me Facebook API kasutada
        soovime
        link: http://clickndrive.ee/juhuslik/, // link kuhu kasutajad
        suunatakse pärast postitusele klikkamist
        picture: data.image, // postituse pilt
        name: data.title, // postituse pealkiri
        description: data.desc // postituse kirjeldus
    };
    function callback(response){ // funktsioon käsitlemaks Facebook'ist
    tulevat vastust
        if (response==null){ // kui vastu saadud andmed puuduvad, siis
        kasutaja ei jaganud postitust oma seinale
            console.log("USER DID NOT SHARE");
        }else{ // kui vastu saadakse post_id, siis kasutaja jagas
        postitust oma seinal.
            console.log("USER SHARED -> ",response.post_id);
        }
        // kohe pärast vastuse saamist soovitakse genereerida uus
        jagatav sisu ette ära, et kasutaja vajutusele saaks kohe reageerida
        $.post("../juhuslik/random.php","share",function(data){ //
        kasutakse jQuery POST funktsiooni random.php failile, et kätte saada
        vajalikud andmed jagamise jaoks.
            fb_share_data = JSON.parse(data) // andmed dekodeeritakse
        })
    }
    FB.ui(obj, callback); // Käivitame Facebook funktsiooni eelpool
    deklareeritud andmetega
}
```

Näide, kuidas saadakse andmebaasist suvaline tsitaat:

```
elseif($fr == 1){ // viide juhuslikule arvule
    $sql = new mysqli($hostname, $Username, $Password, $DbName); //
    luuakse ühendus MySQL andmebaasiga
    if (mysqli_connect_errno()) { // kui ühenduse loomine ebaõnnestub,
    siis katkestatakse skript
        exit();
    }else{ // juhul kui ühendus loodi edukalt
        $sql->set_charset("utf8"); //määratakse ühenduse kodeermise
meetod
        $result = mysqli_query($sql, "SELECT quote, author FROM quotes
ORDER BY RAND() LIMIT 1"); // päring andmebaasi
        if (!$result){ // juhul, kui päring ebaõnnestub
            $sql->close(); // siis andmebaasi ühendus suletakse
            exit(); // ja skriptist väljutakse.
        }else{ // päring õnnestus
            $row = mysqli_fetch_row($result); // võtame esimese
saadaoleva rea päringu vastusest
            echo "<p><b>\". $row[0]. \"</b></p>"; //väljastame
tsitaadi
            echo "<p><i> - ". $row[1]. \"</i></p>"; // väljastame autori
        }
    }
}
```

Google Forms keskkonnas läbi viidud küsitluse ankeet

Kuidas hindate rakenduse kasutuskogemust? (hetkest millal rakenduse lehele sattusite) *

	1	2	3	4	5	
ei saanud midagi aru	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	kõik oli arusaadav

Kuidas hindate rakenduse Random valiku all kuvatavat sisu? *

	1	2	3	4	5	
Üldse ei meeldinud	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Meeldis väga

Kuidas hindate rakenduse Chat valiku all kuvatavat sisu? *

	1	2	3	4	5	
Üldse ei meeldinud	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Meeldis väga

Täpsustavad/muud märkused rakenduse kohta *

Teie vastus
