

# **Paatti-sovellusprojekti**

**Tapio Keränen  
Toni Salminen  
Jari Salokangas  
Lauri Satokangas**

## **Sovellusraportti**

Julkinen  
Versio 0.2.0  
22.5.2012

**Jyväskylän yliopisto  
Tietotekniikan laitos  
Jyväskylä**

<b>Hyväksyjä</b>	<b>Päivämäärä</b>	<b>Allekirjoitus</b>	<b>Nimenselvennys</b>
Projektipäällikkö	__.__.2012		
Tilaaaja	__.__.2012		
Ohjaaja	__.__.2012		

## Tietoa dokumentista

### Tekijät:

- Tapio Keränen (TK) t.tapio.keranen@student.jyu.fi 040-8464981
- Toni Salminen (TS) toni.a.j.salminen@student.jyu.fi 040-5378086
- Jari Salokangas (JS) jari.p.t.salokangas@student.jyu.fi 040-8652014
- Lauri Satokangas (LS) lauri.n.satokangas@student.jyu.fi 040-5735358

**Dokumentin nimi:** Paatti-projekti, Sovellusraportti

**Sivumäärä:** 41

**Tiedosto:** paatti\_sovellusraportti.tex

**Tiivistelmä:** Paatti-projekti kehitti Jyväskylän yliopiston psykologian laitokselle prototyypin tietojärjestelmästä, jolla kuntoutetaan iäkkäitä, mielenterveyspotilaita ja työuupumuksesta kärsiviä sekä tuetaan kyseisiin aihealueisiin liittyvää tutkimusta. Tietojärjestelmässä on kaksi käyttöliittymää, joista tutkijan käyttöliittymällä hallitaan kuntoutettavien kuntoutusohjelmia sekä kuntoutettavan käyttöliittymällä kuntoutettava suorittaa tutkijan laatimaa kuntoutusohjelmaa. Sovellusraportissa kuvataan projektissa toteutettu tietojärjestelmän prototyyppi käyttöliittymän, tietorakenteen, ohjelmointikäytänteiden sekä tavoitteiden toteutumisen osalta.

**Avainsanat:** Arkkitehtuuri, jatkokehitys, komponentit, käytänteet, käyttöliittymä, käyttöohje, rajapinnat, rakenne, sovellus, tavoitteet, testaus, tietojärjestelmä, tietokanta.



## Muutoshistoria

Versio	Päivämäärä	Muutokset	Tekijät
0.0.1	11.5.2012	Dokumentti luotu ja pohjaksi otettu projekti-suunnitelman pohja.	JS
0.0.2	12.5.2012	Muokattu avainsanoja, lukuja "Johdanto" ja "Ohjelmointikäytänteet". Kirjoitettu luvut "Tutkijan käyttöliittymä", "Kuntoutettavan käyttöliittymä", "Sovelluksen rakenne" (osittain), "Tavoitteiden toteutuminen" (osittain) sekä "Yhteenveto".	JS
0.0.3	13.5.2012	Lisätty kuvat lukuihin "Tutkijan sovelluksen näkymäkartta", "Kuntoutettavan sovelluksen näkymäkartta", "Tietojärjestelmän asiakaspalvelinarkkitehtuuri" ja "Sovelluksen tietokanta". Kirjoitettu lukuja "Sovelluksen tietokanta", "Tunnetut puutteet ja virheet" ja "Jatkokehitysideat" sekä muokattu lukua "Toteutusratkaisujen kehittyminen".	JS
0.0.4	14.5.2012	Lisätty kuvat lukuihin "Sovelluksen päänäkymä" ja "Oma profiili -näkyvä". Muokattu lukuja "Sovelluksen tietokanta", "Tunnetut puutteet ja virheet" ja "Jatkokehitysideat".	JS
0.1.0	15.5.2012	Toimitettu versio ohjaajan tarkastettavaksi.	JS
0.1.1	17.5.2012	Muokattu avainsanoja, lukuja "Johdanto", "Termit", "Tutkijan käyttöliittymä", "Kuntoutettavan käyttöliittymä", "Tietojärjestelmän rakenne" (osittain) ja "Yhteenveto".	JS
0.1.2	18.5.2012	Lisätty luku "Tietojärjestelmän tietokanta". Muokattu lukuja "Tietojärjestelmän rakenne" (osittain), "Ohjelmointikäytänteet" ja "Tavoitteiden toteutuminen".	JS
0.1.3	19.5.2012	Poistettu ylimääräisiä termejä. Muokattu lukuja "Tapahtumien luonti ja muokkaus", "Tietojärjestelmän tietokanta".	JS
0.1.4	20.5.2012	Muokattu lukua "Jatkokehitysideat".	JS
0.1.5	21.5.2012	Muokattu sovellusraporttia kokonaisvaltaisesti. Kirjoitettu lukuja "Tietojärjestelmän rakenne ja rajapinnat" ja "Tietojärjestelmän komponentit".	JS

<b>Versio</b>	<b>Päivämäärä</b>	<b>Muutokset</b>	<b>Tekijät</b>
0.1.6	22.5.2012	Lisätty kuvat lukuihin "Sovelluksen näkymien rakenne (kuntoutettava)" ja "Tietojärjestelmän rakenne ja rajapinnat".	JS
0.2.0	22.5.2012	Toimitettu versio ohjaajan tarkastettavaksi.	JS

## Tietoa projektista

Paatti-projekti kehitti Jyväskylän yliopiston psykologian laitokselle prototyypin tietojärjestelmästä, jolla tuetaan vanhusten, mielenterveyspotilaiden ja työuupumuksesta kärsivien itsenäistä elämää ja kuntoutumista, sekä tuetaan kyseisiin aihealueisiin liittyvän tutkimuksen tekemistä.

### Tekijät:

- Tapio Keränen (TK) t.tapio.keranen@student.jyu.fi 040-8464981
- Toni Salminen (TS) toni.a.j.salminen@student.jyu.fi 040-5378086
- Jari Salokangas (JS) jari.p.t.salokangas@student.jyu.fi 040-8652014
- Lauri Satokangas (LS) lauri.n.satokangas@student.jyu.fi 040-5735358

### Tilaaaja:

- Jukka Kaartinen jukka.kaartinen@jyu.fi 0400-248148
- Päivi Lappalainen paivi.k.lappalainen@psyka.jyu.fi 040-8054192
- Raimo Lappalainen raimo.lappalainen@psyka.jyu.fi 050-4432349
- Heikki Lyytinen heikki.lyytinen@psyka.jyu.fi 050-5524892

### Ohjaajat:

- Matti Lehtinen matti.k.lehtinen@jyu.fi 041-4564118
- Jukka-Pekka Santanen santanen@mit.jyu.fi 040-8053299
- Outa Valkama outa.j.valkama@gmail.com 045-3490910

### Yhteystiedot:

- Sähköpostilistat: paatti@korppi.jyu.fi,  
paatti\_opetus@korppi.jyu.fi
- Sähköpostiarkistot: <https://korppi.jyu.fi/list-archive/paatti/>,  
[https://korppi.jyu.fi/list-archive/paatti\\_opetus/](https://korppi.jyu.fi/list-archive/paatti_opetus/)
- Työhuone: Agora C222.2, puh. 040-8053308





# Sisältö

<b>1</b>	<b>Johdanto</b>	<b>1</b>
<b>2</b>	<b>Termit</b>	<b>3</b>
2.1	Aihealueen termit . . . . .	3
2.2	Vaativuusmäärittelyn termit . . . . .	4
2.3	Tekniikat ja kehitysvälineet . . . . .	5
<b>3</b>	<b>Tutkijan käyttöliittymä</b>	<b>6</b>
3.1	Sovelluksen näkymäkartta . . . . .	6
3.2	Sovelluksen näkymien rakenne . . . . .	7
3.3	Sovelluksen päänäkymä . . . . .	7
3.4	Käyttäjien hallinta . . . . .	7
3.5	Ryhmien hallinta . . . . .	8
3.6	Tutkimusten hallinta . . . . .	9
3.7	Tapahtumien luonti ja muokkaus . . . . .	10
3.8	Kuntoutusohjelmien luonti ja muokkaus . . . . .	11
3.9	Jatkokehitykseen sovitut näkymät . . . . .	12
<b>4</b>	<b>Kuntoutettavan käyttöliittymä</b>	<b>13</b>
4.1	Sovelluksen näkymäkartta . . . . .	13
4.2	Sovelluksen näkymien rakenne . . . . .	14
4.3	Sovelluksen päänäkymä . . . . .	15
4.4	Oma profiili -näkymä . . . . .	16
4.5	Edistymisen näkymä . . . . .	17
4.6	Tapahtumat-näkymä . . . . .	17
4.7	Tapahtumien suorittaminen . . . . .	17
4.8	Harjoitteet-näkymä . . . . .	18
4.9	Jatkokehitykseen sovitut näkymät . . . . .	18
<b>5</b>	<b>Tietojärjestelmän rakenne</b>	<b>19</b>
5.1	Tietojärjestelmän asiakas-palvelinarkkitehtuuri . . . . .	19
5.2	Tietojärjestelmän rakenne ja rajapinnat . . . . .	20
5.3	Tietojärjestelmän komponentit . . . . .	20
5.4	Toteutusratkaisujen kehittyminen . . . . .	22

<b>6</b>	<b>Tietojärjestelmän tietokanta</b>	<b>23</b>
6.1	Tietokannan ER-kaavio . . . . .	23
6.2	USER-taulu . . . . .	25
6.3	USERGROUP-taulu . . . . .	25
6.4	ROLE-taulu . . . . .	25
6.5	RESEARCH-taulu . . . . .	25
6.6	SCHEDULE-taulu . . . . .	26
6.7	EVENT-taulu . . . . .	26
6.8	EVENTTIME-taulu . . . . .	26
6.9	EVENTTYPE-taulu . . . . .	27
6.10	TASK-taulu . . . . .	27
6.11	TASKTYPE-taulu . . . . .	28
6.12	CHOICE-taulu . . . . .	28
6.13	TASKDATA-taulu . . . . .	28
6.14	BINARYDATA-taulu . . . . .	28
6.15	TEXTDATA-taulu . . . . .	29
<b>7</b>	<b>Ohjelmointikäytännöt</b>	<b>30</b>
7.1	Lähdekoodin muotoilu ja tiedostojen nimeäminen . . . . .	30
7.2	Lähdekoodin kommentointi . . . . .	30
7.3	Lähdekoodiesimerkki . . . . .	30
7.4	Kehitysympäristö . . . . .	31
7.5	Testauksen käytännöt . . . . .	31
7.6	Testauksen tulokset . . . . .	32
<b>8</b>	<b>Tavoitteiden toteutuminen</b>	<b>33</b>
8.1	Vaatimusten toteutuminen . . . . .	33
8.2	Tunnetut puutteet ja virheet . . . . .	34
8.3	Jatkokehitysideat . . . . .	36
<b>9</b>	<b>Käyttöohjeita ylläpitäjälle</b>	<b>39</b>
9.1	Sovelluksen asennus palvelimelle . . . . .	39
9.2	Tietokokonaisuuden hallintanäkymän lisääminen . . . . .	39
9.3	Tehtäväkomponenttien lisääminen . . . . .	39
<b>10</b>	<b>Yhteenveto</b>	<b>40</b>
<b>11</b>	<b>Lähteet</b>	<b>41</b>

# 1 Johdanto

Paatti-projekti toteutti Sovellusprojekti-kurssilla keväällä 2012 Jyväskylän yliopiston psykologian laitokselle prototyypin tietojärjestelmästä, jolla tuetaan kuntoutettavien itsenäistä elämää ja kuntoutumista sekä tutkijoiden tekemää tutkimusta. Kuntoutettavaan kuuluu muun muassa iäkkäitä, työuupumuksesta kärsiviä ja mielenterveyspotilaita. Tietojärjestelmällä voidaan tukea tutkimusta ja parantaa kuntoutettavien hyvinvointia monin eri tavoin.

Iäkkäitä ihmisiä halutaan auttaa elämään tervettä ja itsenäistä elämää omassa kodissaan pidempään. Iäkkäillä on usein muistihäiriöitä, jotka estävät heitä noudattamasta hoito-ohjeita. Tilaaja haluaa pystyä muistuttamaan kuntoutettavia hoito-ohjeiden noudattamisesta. Useasti varttuneempien ihmisten fyysinen aktiivisuus alenee, ja tämä johtaa toimintakyvyn heikkenemiseen, joka estää heitä elämästä itsenäisesti omassa kodissaan. Tilaaja haluaa motivoida ikäihmisiä liikkumaan enemmän oma-toimisesti.

Työuupumuksesta kärsivät eivät useasti muista pitää taukoja elämässään ja työssään. Heitä pitää muistuttaa rentoutusharjoituksista, jotka helpottavat stressiä, auttavat toipumaan ja opettavat keinoja selviytyä uupumuksesta.

Mielenterveyspotilaiden hoito vaatii paljon hoitohenkilökunnan läsnäoloa. Läsnäolo ei kuitenkaan ole aina mahdollista, kun hätä on suurin. Potilaan tilaa tulee voida seurata tapaamisten välillä, sekä potilaan ja tutkijan tulee tarvittaessa saada yhteys toisiinsa tapaamisten välillä.

Tilaaja haluaa joustavan yhteyden kuntoutettavan, omaisen ja tutkijan välille. Tutkijan ja kuntoutettavan välille halutaan yhteys, joka mahdollistaa rutiiniluonteisten tehtävien suorittamisen ilman tutkijan läsnäoloa ajasta ja paikasta riippumattomasti. Yhteys mahdollistaa myös nopeamman viestinnän kuntoutettavan ja tutkijan välillä sekä antaa mahdollisuuden tutkijalle ja omaiselle seurata kuntoutukseen liittyvien tapahtumien suorittamista lähes reaaliajassa.

Projektissa toteutettu tietojärjestelmä koostuu kahdesta käyttöliittymästä: tutkijan ja kuntoutettavan käyttöliittymästä. Tutkijan käyttöliittymä on suunniteltu tutkijoiden ja omaisten käytettäväksi. Käyttöliittymällä voi hallita käyttäjiä, ryhmiä ja tutkimuksia sekä luoda kuntoutusohjelmia ryhmille. Kuntoutettavan käyttöliittymä on suunniteltu kuntoutettavien käyttöön ja sillä voi suorittaa kuntoutusohjelmia sekä tarkastella suorittamia kuntoutusohjelman tapahtumia. Toteutetut käyttöliittymä-

prototyypit ovat yhteydessä palvelimella olevaan tietokantaan.

Paatti-projektissa toteutettu tietojärjestelmän prototyyppi ei sisällä kaikkia vaadittuja toimintakokonaisuuksia. Tilaajien kanssa jatkokehitykseen sovittiin tutkijan käyttöliittymästä mediatiedostojen lataus ja esikatselunäkymä, suoritiedatan graafiseen analysointiin käytettävä raporttinäkymä, viestien lukemiseen ja lähettämiseen tarkoitettu viestinäkymä sekä ohjenäkymä. Kuntoutettavan käyttöliittymästä tilaajien kanssa jatkokehitykseen sovittiin päiväkirjamerkintöjen tallentamiseen ja lukemiseen tarkoitettu päiväkirjanäkymä sekä viestien lukuun tarkoitettu viestinäkymä.

Sovellusraportissa kuvataan vaatimusmäärittelyn [1] toiminnallisten vaatimuksien toteutustapa ja toteutuminen. Lisäksi raportissa kuvataan sovelluksen käyttöliittymiä, yleisiä toteutusratkaisuja, toteutuksen tietorakennetta ja yhteyksiä tiedostoihin, puutteellisia toteutusratkaisuja, suoritettuja testauksia sekä jatkokehitysideoita. Projektin tuloksia, käytänteitä, tehtävien työmääriä, työnjakoa, aikataulua, sekä riskejä ja muita projektin läpivientiin liittyviä asioita käsitellään projektiraportissa [6].

Luvussa 2 kuvataan aihealueeseen ja sovellukseen liittyvät termit ja niiden merkitys. Luvuissa 3 ja 4 kuvataan tutkijan ja kuntoutettavan käyttämien sovellusten käyttöliittymien näkymät ja toiminnot. Luvussa 5 kuvataan järjestelmän rakenne ja toteutusratkaisuja. Luvussa 6 kuvataan järjestelmän tietokanta ja tietokannassa olevien taulujen merkitykset. Luvussa 7 kuvataan projektin ohjelmointikäytänteitä sekä testauskäytänteitä ja testauksen tuloksia. Luvussa 8 kuvataan projektin tavoitteiden toteutumista, järjestelmän tunnettuja virheitä ja puutteita sekä projektin aikana esiintulleita jatkokehitysideoita. Luvussa 9 sisältää käyttöohjeita sovelluksen ylläpitäjille.

## 2 Termit

Luvussa kuvataan projektissa käytettäviä aihealueen termejä sekä tekniikoita ja kehitysvälineitä.

### 2.1 Aihealueen termit

Projektin sovellusraportin aihealueen termejä ovat seuraavat:

<b>Hälytystehtävä</b>	on tehtävätyyppi, joka lähettää järjestelmän kautta viestin toiselle käyttäjälle.
<b>Kuntoutettava</b>	on tietojärjestelmän käyttäjä, joka suorittaa kuntoutusohjelmia.
<b>Kuntoutusohjelma</b>	on tutkijan ryhmälle tai käyttäjälle laatima kokonaisuus aikataulutettuja tapahtumia.
<b>Käyttäjärooli</b>	on käyttäjän käyttöoikeuksia tietojärjestelmässä kuvaava rooli. Niitä ovat mm. kuntoutettava ja tutkija. 1
<b>Mobiililaite</b>	on älypuhelin tai sormitietokone.
<b>Ohje</b>	on tehtävään liittyvä kuvaus teksti-, ääni-, kuva- tai video-muodossa.
<b>Omainen</b>	on käyttäjä, jolla on oikeus seurata kuntoutettavan edistymistä ja aktiivisuutta.
<b>Ryhmä</b>	on käyttäjistä koostuva kokonaisuus, jolle mm. voidaan laatia kuntoutusohjelma.
<b>Suorite</b>	on tehtävän suorittamisesta tietokantaan tallennettu data.
<b>Tapahtuma</b>	on tutkijan laatima tehtävien kokonaisuus, joka liitetään osaksi kuntoutusohjelmaa.
<b>Tehtävä</b>	on tapahtuman sisällä oleva yksittäinen tehtävä, joka voi sisältää mm. tekstiä, ääntä, kuvan tai videon.

<b>Tutkija</b>	on käyttäjä, joka laatii kuntoutusohjelmia kuntoutettaville ja seuraa niiden toteutumisia.
<b>Tutkijan käyttöliittymä</b>	on käyttöliittymä, jolla tutkija luo tapahtumia, aikatauluttaa tapahtumia, hallitsee käyttäjien ja ryhmien tietoja sekä muodostaa suoritteista erilaisia raportteja.
<b>Tutkimus</b>	on samaan tutkimusaiheeseen tai -kohteeseen liittyvien ryhmien kokonaisuus.

## 2.2 Vaatimusmäärittelyn termit

Vaatimusmäärittelyssä esiintyviä termejä ovat seuraavat:

<b>Etenemisvaihtoehto</b>	on monivalintatehtävän yksi valinta.
<b>Hälytyskomponentti</b>	on tehtävä, joka lähettää profiiliviestin komponentin asetuksiin määritetyille henkilöille.
<b>Käyttäjaprofiili</b>	on järjestelmässä olevan käyttäjän profiili, joka sisältää tunnistetietoja ja käyttöoikeuksia.
<b>Mediasisältö</b>	koostuu tietojärjestelmässä olevista multimediatiedostoista, joita voidaan sisällyttää tehtäviin.
<b>Muistutus</b>	huomauttaa käyttäjää tapahtuneesta asiasta. Muistutus voi olla ääni-, värinä- ja tekstimuodossa.
<b>Sisältökomponentti</b>	on tehtävä, joka esittää käyttäjälle informaatiota, kuten tekstiä, ääntä, valokuvan tai videon.
<b>Syötekomponentti</b>	on tehtävä, joka tallentaa järjestelmään käyttäjän antamaa informaatiota, kuten tekstiä, ääntä, valokuvan tai videon.
<b>Tapahtumamuokkain</b>	on tutkijan käyttöliittymässä oleva kuntoutusohjelmissä käytettävien tapahtumien laatimiseen käytettävä työkalu.
<b>Tapahtumatyyppi</b>	on tapahtumalle asetettu tyyppi, jolla tapahtuman sisältö voidaan tunnistaa. Niitä voivat olla mm. liikunta- tai rentoutustapahtumat.

**Valintakomponentti** on tehtävä, johon käyttäjä vastaa valitsemalla vaihtoehdon.

## 2.3 Tekniikat ja kehitysvälineet

Dokumentissa esiintyviä tekniikoita ja kehitysvälineitä ovat seuraavat:

<b>CSS</b>	on WWW-dokumenteille kehitetty tyylikieli.
<b>Git</b>	on hajautettu versiohallintajärjestelmä.
<b>HTML5</b>	on uusi versio WWW-sivujen tekemiseen käytetystä HTML-kuvauskielestä.
<b>JavaDoc</b>	on ohjelma, jonka avulla Javan lähdekoodista voidaan generoida luokkadokumentaatio.
<b>JavaScript</b>	on WWW-ympäristössä käytettävä komentosarjakieli, jolla voidaan mm. lisätä WWW-sivuille dynaamista toiminnallisuutta.
<b>NetBeans</b>	on integroitu ohjelmointiympäristö mm. Java- ja JavaScript-ohjelmointikielille.
<b>Vaadin</b>	on Java-sovelluskehitysympäristö WWW-sovellusten luomiseen.
<b>YouSource</b>	on Git-versiohallintaohjelmistoa tukeva lähdekoodien julkistusjärjestelmä, jota käytetään WWW-käyttöliittymällä.

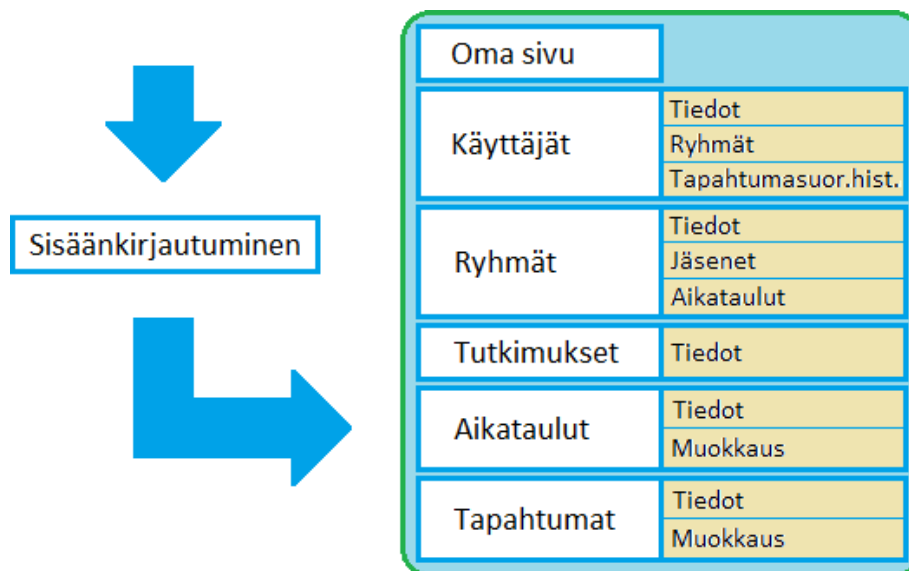
### 3 Tutkijan käyttöliittymä

Luvussa esitellään pääosin tutkijan käytettäväksi toteutetun sovelluksen käyttöliittymän eri näkymät sekä niihin liittyvät toiminnot ja niiden väliset suhteet.

Toteutettu tutkijan käyttöliittymä ei sisällä kaikkia tarvittavia toiminnallisuuksia. Toteutettuihin toiminnallisuuksiin kuuluu sovelluksen sisäänkirjautuminen *Sisäänkirjautuminen*-näkyssä, käyttäjien luonti ja muokkaus *Käyttäjät*-näkyssä, ryhmien luonti ja muokkaus *Ryhmät*-näkyssä, tutkimusten luonti ja muokkaus *Tutkimukset*-näkyssä, tapahtumien luonti ja muokkaus *Tapahtumat*-näkyssä sekä kuntoutusohjelmien luonti ja muokkaus *Kuntoutusohjelmat*-näkyssä. Jatkokehityksessä toteutetaan multimediatiedostojen lataaminen ja esikatselu *Mediasisältö*-näkyssä, suoritedatasta luotavien raporttien luonti *Raportit*-näkyssä, viestien lähettäminen ja vastaanottaminen *Viestit*-näkyssä sekä ohjeiden luku *Ohje*-näkyssä.

#### 3.1 Sovelluksen näkymäkartta

Kuvassa 3.1 havainnollistetaan tutkijan käyttöliittymän näkymien välistä jakautumista eri näkymiin ja niiden välilehtiä näkymäkarttana. Luvuissa 3.3 - 3.9 kuvataan kunkin näkymän toiminnallisuus ja yhteydet muihin näkymiin.



Kuva 3.1: Tutkijan käyttöliittymän näkymäkartta.



## 3.2 Sovelluksen näkymien rakenne

Sovelluksen toiminnallisuus on sijoitettu välilehtinä esitettäviin näkymiin. Välilehtiratkaisu mahdollistaa sulavan siirtymisen näkymästä toiseen yläreunan välilehtikuvakkeiden kautta. Muista navigointimahdollisuuksista näkymien välillä on mainintoja luvuissa 3.4 - 3.8.

Näkymien rakenne on ulkoasultaan ja sijoittelultaan samanlainen. Vasemmalla puolella näkymää on hakukomponentti, jolla käyttäjä voi hakea tiettyyn näkymään liittyviä tietoja. Esimerkiksi käyttäjien hallintanäkymässä kyseisellä hakukomponentilla voi hakea tietokannasta järjestelmässä olevia käyttäjiä.

Näkymien oikealla puolella olevassa välilehdessä näytetään hakutulostasta valittuun tietokokonaisuuteen liittyviä tietoja, missä voidaan muokata kyseiseen tietokokonaisuuteen liittyviä tietoja.

## 3.3 Sovelluksen päänäkymä

Sovelluksen päänäkymänä toimii *Oma sivu* -näkö, joka näyttää tutkijalle ensiarvoista tietoa kuntoutettavista. Näkö myös huomauttaa tutkijalle, jos hänellä on jokin asia jäänyt kesken aiemmilla istunnoilla hänen kirjautuessaan ulos järjestelmästä. Kyseinen näkö avautuu aina ensimmäisenä, kun sovellukseen kirjaudutaan sisään. Tämän näkökän kautta kirjaudutaan myös ulos järjestelmästä.

Sovelluksen päänäkökän rakenne eroaa ulkoasultaan muista näkömistä, sillä päänäkökän ei luoda tai muokata järjestelmän tietokannan tietoja, vaan päänäkökän toimii ensisijaisena informaatiovirtana sovelluksen käyttäjälle.

Näkökän liittyvät toiminnallisuudet sovittiin tilaajien kanssa projektin jälkeiseen jatkokehitykseen. Uloskirjautumisen sijainnin muuttamista ja muita jatkokehitysideoita kuvataan luvussa 8.3.

## 3.4 Käyttäjien hallinta

Käyttäjien hallintanäkökän kuvassa ?? tutkija voi luoda uusia käyttäjiä järjestelmään, hakea olemassa olevia käyttäjiä ja muokata heidän tietojan, hallita käyttäjien

ryhmiä sekä tarkastella käyttäjän suorittamien tapahtumien suoritedataa. Päänäkymän toimintakokonaisuudet ovat uuden käyttäjän luonti ja haetun käyttäjän tietojen muokkaaminen. Tietojen muokkaamisen osatoimintakokonaisuudet on jaettu alivälilehdelle ja ne ovat käyttäjän perustietojen muokkaaminen, käyttäjän ryhmien jäsenyyden tarkasteleminen, lisääminen ja poistaminen sekä tapahtumasuoritushistoria tarkastelu ja vieminen.

**Luotaessa uutta käyttäjää** tutkija painaa hakukomponentin *Luo uusi* -painiketta, jolloin oikealla oleva alivälilehti aktivoituu muokattavaksi. Tämän jälkeen tutkija syöttää pakollisiin syötekenttiin tarvittavat käyttäjän perustiedot, jonka jälkeen hän voi tallentaa käyttäjän järjestelmän tietokantaan. Tutkija voi tallentamisen jälkeen jatkaa juuri tallennetun käyttäjän muokkausta tai valita toisen käyttäjän muokattavaksi.

**Käyttäjän tietoja muokattaessa** tutkija voi muokata käyttäjän perustietoja *Käyttäjän tiedot* -alivälilehdessä. Muokattaviin käyttäjän perustietoihin kuuluvat etunimi, sukunimi, käyttäjätunnus ja käyttäjän kuvaus. Välilehdellä voidaan myös tallentaa tai poistaa kyseinen muokattavaksi valittu käyttäjä *Poista käyttäjä* -painikkeella. *Käyttäjän ryhmät* -välilehdellä tutkija voi nähdä ryhmät, joissa käyttäjä on jäsenenä ja voi halutessaan lisätä käyttäjän uuteen ryhmään tai poistaa käyttäjän valitusta ryhmästä. Näkymästä voi myös päästä tarkastelemaan valitun käyttäjän ryhmän tietoja *Tarkastele valittua ryhmää* -painikkeen avulla. *Suoritetut tapahtumat* -välilehdellä tutkija voi tarkastella käyttäjän suorittamista tapahtumista kerättyä suoritedataa ja ottaa kyseisen datan xls-tiedostomuodossa Excel-taulukkolaskentaohjelmaan painamalla *Vie .xls-muotoon* -painiketta.

### 3.5 Ryhmien hallinta

Ryhmien hallintanäkymän kuvassa ?? tutkija voi luoda uusia ryhmiä järjestelmään, hakea olemassa olevia ryhmiä ja muokata niiden tietoja, hallita ryhmään kuuluvia käyttäjiä sekä hallita ryhmien kuntoutusohjelmia. Näkymän päätoimintakokonaisuudet ovat uuden ryhmän luonti ja haetun ryhmän tietojen muokkaaminen. Alivälilehdille jaetut tietojen muokkaamisen osatoimintakokonaisuudet ovat ryhmän perustietojen muokkaaminen, ryhmän käyttäjien jäsenyyksien tarkasteleminen, lisääminen ja poistaminen sekä ryhmän kuntoutusohjelmien tarkasteleminen, lisääminen ja poistaminen.

**Luotaessa uutta ryhmää** tutkija painaa hakukomponentin *Luo uusi* -painiketta, jolloin oikealla oleva alivälilehti aktivoituu muokattavaksi. Tämän jälkeen tutkija syöt-

tää pakollisiin syötekenttiin tarvittavat ryhmän perustiedot, jonka jälkeen hän voi tallentaa ryhmän järjestelmän tietokantaan. Tutkija voi tallentamisen jälkeen jatkaa juuri tallennetun ryhmän muokkausta tai valita toisen ryhmän muokattavaksi.

**Ryhmän tietoja muokattaessa** tutkija voi muokata valitun ryhmän perustietoja *Ryhmän tiedot* -välilehdellä. Muokattaviin ryhmän perustietoihin kuuluvat nimi, ryhmän rooli, ryhmään liitetty tutkimus ja ryhmän kuvaus. Välilehdellä voidaan myös tallentaa tai poistaa kyseinen muokattavaksi valittu ryhmä *Poista ryhmä* -painikkeella. *Ryhmän jäsenet* -välilehdellä tutkija voi nähdä ryhmään kuuluvat käyttäjät ja voi halutessaan joko lisätä ryhmään uuden käyttäjän tai poistaa ryhmästä valitun käyttäjän. Välilehdeltä voi myös päästä suoraan tarkastelemaan valitun ryhmän jäsenenä olevan käyttäjän tietoja *Tarkastele valittua ryhmää* -painikkeen avulla. *Ryhmän aikataulut* -välilehdellä tutkija voi tarkastella ryhmän sisältämiä kuntoutusohjelmia, lisätä uusia kuntoutusohjelmia ryhmälle sekä poistaa ryhmältä kuntoutusohjelmia. Näkymästä voi hypätä myös tarkastelemaan ryhmällä olevian kuntoutusohjelman tietoja *Tarkastele valittua aikataulua* -painikkeen avulla.

### 3.6 Tutkimusten hallinta

Tutkimusten hallintänäköymän kuvassa ?? tutkija voi luoda uusia tutkimuksia järjestelmään sekä hakea ja muokata olemassa olevia tutkimuksia. Näköymän päätoimintakokonaisuudet ovat uuden tutkimuksen luonti sekä haetun tutkimuksen tietojen muokkaaminen.

**Luotaessa uutta tutkimusta** tutkija painaa hakukomponentin *Luo uusi* -painiketta, jolloin oikealla oleva alivälilehti aktivoituu muokattavaksi. Tämän jälkeen tutkija syöttää pakollisiin syötekenttiin tarvittavat tutkimuksen perustiedot, jonka jälkeen hän voi tallentaa tutkimuksen järjestelmän tietokantaan. Tutkija voi tallentamisen jälkeen jatkaa juuri tallennetun tutkimuksen muokkausta tai valita toisen tutkimuksen muokattavaksi.

**Tutkimuksen tietoja muokattaessa** tutkija voi muokata tutkimuksen perustietoja *Tutkimuksen tiedot* -välilehdellä. Muokattaviin tutkimuksen perustietoihin kuuluvat nimi ja tutkimuksen kuvaus. Välilehdellä voidaan myös tallentaa tai poistaa kyseinen muokattavaksi valittu tutkimus *Poista tutkimus* -painikkeella. Välilehdessä näytetään myös tutkimukseen kuuluvat ryhmät, joita voi hypätä tarkastelemaan *Tarkastele valittua ryhmää* -painikkeen avulla.

### 3.7 Tapahtumien luonti ja muokkaus

Tapahtumien luonti- ja muokkausnäkyvän kuvassa ?? tutkija voi luoda uusia tapahtumia järjestelmään sekä hakea ja muokata olemassa olevia tapahtumia. Näkyvän päätoimintakokonaisuudet ovat uuden tapahtuman luonti sekä haetun tapahtuman muokkaaminen. Tapahtuman muokkaamisen osatoimintakokonaisuudet ovat tapahtuman perustietojen muokkaaminen sekä tapahtuman rakenteen ja sisällön muokkaaminen.

**Luotaessa uutta tapahtumaa** tutkija painaa hakukomponentin *Luo uusi* -painiketta, jolloin oikealla oleva alivälilehti aktivoituu muokattavaksi. Tämän jälkeen tutkija syöttää pakollisiin syötekenttiin tarvittavat tapahtuman perustiedot, jonka jälkeen hän voi muokata tapahtuman rakennetta ja sisältöä tapahtumamuokkaimella, joka sijaitsee *Tapahtuman sisältö* -välilehdessä. Tutkija voi tallentaa tapahtuman järjestelmän tietokantaan *Tallenna tapahtuma* -painikkeella. Tutkija voi tallentamisen jälkeen jatkaa juuri tallennetun tapahtuman muokkausta tai valita toisen tapahtuman muokattavaksi.

**Tapahtuman tietoja muokattaessa** tutkija voi muokata tapahtuman perustietoja *Tapahtuman tiedot* -välilehdellä. Muokattaviin tapahtuman perustietoihin kuuluvat otsikko, kuvaus ja tyyppi. Välilehdellä voidaan myös tallentaa tai poistaa kyseinen muokattavaksi valittu tapahtuma *Poista tapahtuma* -painikkeella. *Tapahtuman sisältö* -välilehdellä tutkija voi muokata tapahtuman rakennetta ja sisältöä tapahtumamuokkaimella.

Tapahtumamuokkaimen vasemmalla puolella olevassa **piirtoalueessa** näytetään tapahtuman rakenne. Uusia tehtäväkomponentteja lisätään piirtoalustaan painalla hiiren oikeaa painiketta piirtoalustan tyhjässä tilassa. Tämän jälkeen tutkija valitsee hiirivalikosta lisättäväksi halutun tehtäväkomponenttityypin. Kun tutkija painaa hiiren oikeaa painiketta tehtäväkomponentin sisällä, voi tutkija siirtää tai poistaa valitun tehtäväkomponentin tai luoda etenemistien kahden komponentin välille. Etenemistie kertoo seuraavan tehtävän osoitteen, kun halutaan siirtyä tapahtumaa suoritettaessa tehtävästä toiseen. Tehtävänkomponentin sisältöä voidaan muokata tapahtumamuokkaimen oikean puolen tietoalueessa, jossa näytetään piirtoalueesta hiirellä valittuun tehtäväkomponenttiin liittyviä tietoja.

**Lisättäviä tehtäväkomponenttityyppejä** ovat tekstikomponentti, monivalintakomponentti, liukusäädinkomponentti, sensorikomponentti, mediakomponentti sekä hälytyskomponentti, joista projektissa ehdittiin toteuttaa teksti- ja monivalintakompo-

nentti. Tekstikomponentin muokattaviin ominaisuuksiin kuuluvat otsikko ja tekstisisältö, kun monivalintakomponentin muokattaviin ominaisuuksiin kuuluvat otsikko sekä etenemismvaihtoehtojen lisääminen, etenemismvaihtoehdon seuraavan tapahtuman osoitteen valitseminen, etenemismvaihtoehdon arvon valitseminen sekä etenemismvaihtoehdon tekstin lisääminen.

### 3.8 Kuntoutusohjelmien luonti ja muokkaus

Kuntoutusohjelmien luonti- ja muokkausnäkökuvan kuvassa ?? tutkija voi luoda uusia kuntoutusohjelmia järjestelmään sekä hakea ja muokata olemassa olevia kuntoutusohjelmia. Näkökuvan päätoimintakokonaisuudet ovat uuden kuntoutusohjelman luonti sekä haetun kuntoutusohjelman tietojen muokkaaminen. Tietojen muokkauksen osatoimintakokonaisuudet ovat kuntoutusohjelman perustietojen muokkaaminen sekä kuntoutusohjelmassa olevien tapahtumien tarkasteleminen, lisääminen, poistaminen ja aikatauluttaminen.

**Luotaessa uutta kuntoutusohjelmaa** tutkija painaa hakukomponentin *Luo uusi* -painiketta, jolloin oikealla oleva alivälilehti aktivoituu muokattavaksi. Tämän jälkeen tutkija syöttää pakollisiin syötekenttiin tarvittavat kuntoutusohjelman perustiedot, jonka jälkeen hänen täytyy lisätä tapahtumia kuntoutusohjelmaan *Kuntoutusohjelman muokkaaminen* -välilehdellä. Siinä tutkija voi lisätä järjestelmässä olevia tapahtumia kuntoutusohjelmaan, poistaa kuntoutusohjelmasta tapahtumia sekä aikatauluttaa kuntoutusohjelmaan lisäämiään tapahtumia. Tutkija voi tallentamisen jälkeen jatkaa juuri tallennetun kuntoutusohjelman muokkausta tai valita toisen kuntoutusohjelman muokattavaksi.

**Kuntoutusohjelman tietoja muokattaessa** tutkija voi muokata valitun kuntoutusohjelman perustietoja *Aikataulun tiedot* -välilehdellä. Muokattaviin kuntoutusohjelman perustietoihin kuuluu kuntoutusohjelman kuvaus. Välilehdellä voidaan myös tallentaa tai poistaa kyseinen muokattavaksi valittu kuntoutusohjelma. *Aikataulun muokkaus* -välilehdellä tutkija voi tarkastella kuntoutusohjelman sisältämiä tapahtumia ja niiden ajoitusta, lisätä uusia tapahtumia kuntoutusohjelmaan ja aikatauluttaa ne sekä poistaa tapahtumia kuntoutusohjelmasta.

Tutkija voi myös **luoda vapaaehtoisia kuntoutusohjelmia** *Aikataulun muokkaus* -välilehdellä. Tällöin kuntoutusohjelmassa oleva tapahtuma asetetaan vapaaehtoiseksi *Merkitse tapahtuma vapaaehtoiseksi harjoitteeksi* -asetusnapilla.

### 3.9 Jatkokehitykseen sovitut näkymät

Tilaajan kanssa jatkokehitykseen sovitut näkymät ovat mediatiedostojen lataamis- ja tarkastelunäkymä, raporttinäkymä sekä ohjenäkymä. **Mediatiedostojen lataamis- ja tarkastelunäkymässä** tutkija voi lisätä mediatiedostoja järjestelmään ja esikatsella järjestelmässä olevia mediatiedostoja. Mediatiedostoilla tarkoitetaan kuva-, ääni- tai videotiedostoja. Mediatiedostoja voi järjestelmään lisäämisen jälkeen käyttää mediakomponentin (katso luku 3.7) näytettävänä sisältönä. **Raporttinäkymässä** tutkija voi tarkastella järjestelmään tallentunutta suoritettua graafisessa muodossa sekä luoda erilaisia raportteja eri tiedostoformaatteihin, esimerkiksi pdf-muotoon. **Ohjenäkymässä** tutkija voi lukea sovelluksen käyttöohjeita. Mahdollisesti kyseinen ohjenäkymä saattaa olla järkevä avautua omaan selainikkunaan, eikä täten välttämättä tarvitsisi sijaita välilehtenä.

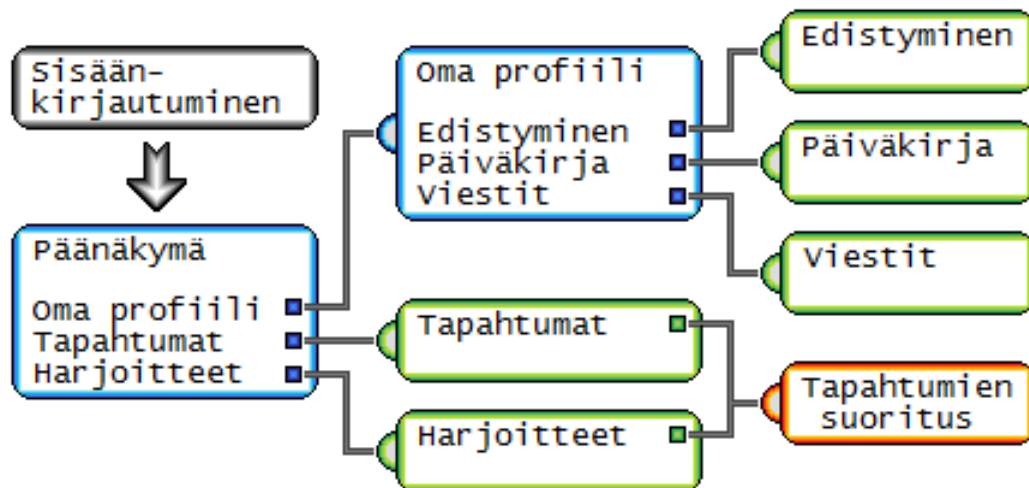
## 4 Kuntoutettavan käyttöliittymä

Luvussa esitellään pääosin kuntoutettavan käytettäväksi toteutetun sovelluksen käyttöliittymän eri näkymät sekä niihin liittyvät toiminnot ja niiden väliset suhteet.

Toteutettu kuntoutettavan käyttöliittymä ei sisällä kaikkia tarvittavia toiminnallisuksia. Toteutettuihin toiminnallisuuksiin kuuluu sovelluksen sisäänkirjautuminen *Sisäänkirjautuminen*-näkyssä, edistymisen seuraaminen *Edistyminen*-näkyssä, tapahtumien suorittaminen *Tapahtumat*-näkyksen kautta sekä vapaaehtoisten tapahtumien suorittaminen *Tutkimukset*-näkyssä. Jatkokehityksessä toteutetaan päiväkirjaviestien luonti ja lukeminen *Päiväkirja*-näkyssä sekä ryhmäkohtaisten viestien lukeminen *Viestit*-näkyssä.

### 4.1 Sovelluksen näkymäkartta

Kuvassa 4.1 havainnollistetaan kuntoutettavan käyttöliittymän näkymien väliset suhteet näkymäkarttana. Luvuissa 4.3 - 4.8 kuvataan kunkin näkymän toiminnallisuus ja yhteydet muihin näkymiin.

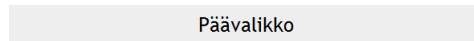


Kuva 4.1: Kuntoutettavan käyttöliittymän näkymäkartta.

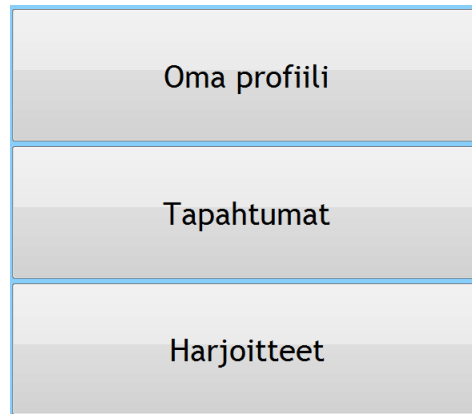
## 4.2 Sovelluksen näkymien rakenne

Kuntoutettavan käyttöliittymän näkymien rakenne ja ulkoasu noudattaa samaa tyyliä koko sovelluksessa. Jokainen näkymä on jaettu kolmeen osaan. Ylimpänä on kaistale otsikkotiedolle (kuva 4.2), jossa näytetään esimerkiksi valikoissa valikon nimi sekä tapahtumaa suoritettaessa tapahtuman nimi. Näkymämallin keskellä olevassa sisältöalueessa (kuva 4.3) näytetään valikoissa navigointipainikkeet, alavalikoissa valikon sisältö ja tapahtumaa suoritettaessa tapahtuman tehtävän sisältö.

Näkymämallissa alimpana on varattu alue toimintopainikkeille (kuva 4.4). Vasen toimintopainike toimii aina takaisin-, peruutus- tai kieltopainikkeena, kun keskimmäinen ja oikea painike toimivat eri tavalla eri näkymissä. Esimerkiksi tapahtumaa suoritettaessa keskimmäinen ja oikea painike toimivat tapahtuman tehtävien etenemis- ja paluupainikkeina. Oikeaa painiketta käytetään myös hyväksymispainikkeena tietyissä tilanteissa, kuten esimerkiksi tapahtuman keskeyttämisessä.



Kuva 4.2: Käyttöliittymän otsikkoalue.



Kuva 4.3: Käyttöliittymän sisältöalue.

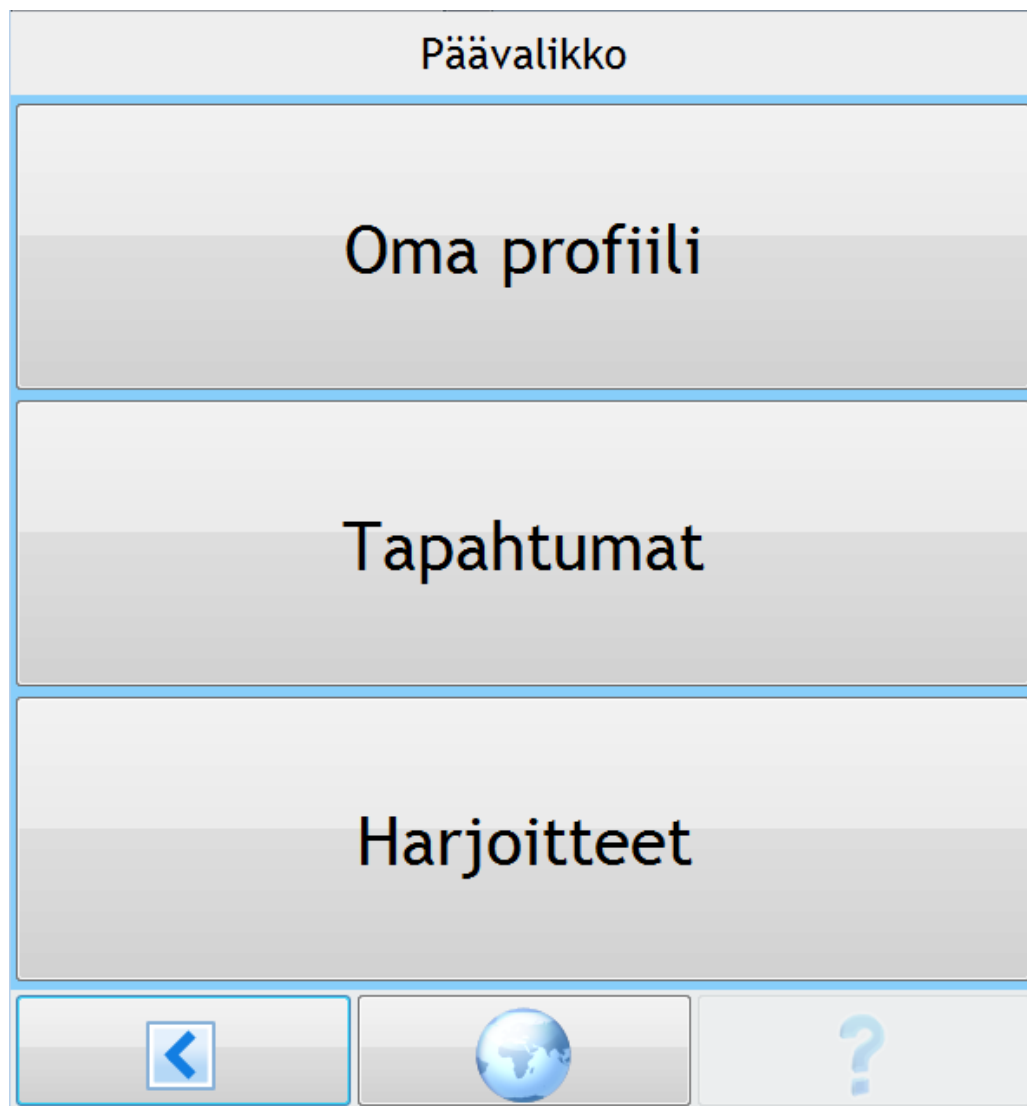


Kuva 4.4: Käyttöliittymän navigointipainikealue.



### 4.3 Sovelluksen päänäkymä

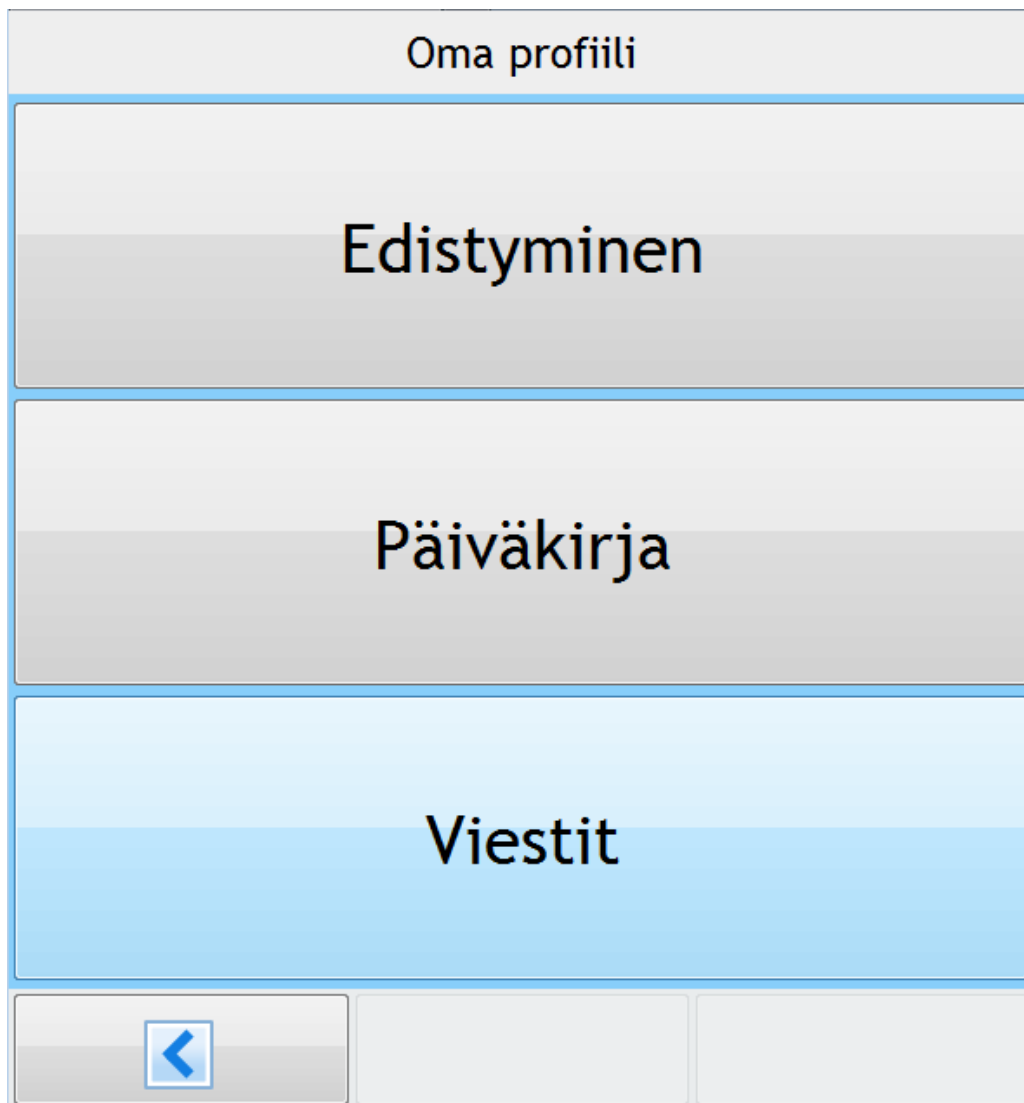
Sovelluksen päänäkymästä (katso kuva 4.2) pääsee *Oma profiili*, *Tapahtumat* ja *Harjoitteet*-näkyymiin vastaavilla painikkeilla. Päävalikon vasemmalla painikkeella kuntoutettava voi kirjautua ulos sovelluksesta ja keskipainikkeella kuntoutettavalle avautuu näkymä, jossa näytetään sovellukseen liittyviä tietoja.



Kuva 4.5: Sovelluksen päävalikko.

## 4.4 Oma profiili -näkyvä

*Oma profiili* -näkyvän kautta pääsee *Edistyminen*, *Päiväkirja* sekä *Viestit*-näkyisiin vastaavilla painikkeilla. Päävalikon vasemmalla painikkeella kuntoutettava voi palata takaisin luvun 4.3 päänäkymään. Muilla alareunan painikkeilla ei ole toiminnallisuutta.



Kuva 4.6: Oma profiili -alavalikko.

## 4.5 Edistymisen-näkymä

Kuntoutettava pääsee *Edistymisen*-näkömään luvun 4.4 näkömän kautta. Näkömäsä kuntoutettava voi tarkastella suorittamiaan tapahtumia. Päävalikon vasemmalla painikkeella kuntoutettava voi palata takaisin luvun 4.4 näkömään. Muilla alareunan painikkeilla ei ole toiminnallisuutta.

## 4.6 Tapahtumat-näkymä

Kuntoutettava pääsee *Tapahtumat*-näkömään luvussa 4.3 esitetyn päänäkömän kautta. Näkömässä näytetään kuntoutettavalle hänen suorittamattomien tapahtumien lista. Kuntoutettava voi valita tapahtumalistasta suoritettavaksi haluamansa tapahtuman, jos se on ajoituksen osalta suoritettavissa sillä hetkellä. Tapahtuman suorittamiseen siirrytään painamalla lista-alkion oikealla puolella olevaa *Aloita*-painiketta.

Kuntoutettava voi alareunan vasemmalla painikkeella palata takaisin päävalikkoon sekä muilla alareunan painikkeilla voidaan valita näkömään näytettäväksi joko tulevia tai rästiin jääneitä tapahtumia, jolloin painettu painike painautuu pohjaan. Painamalla painettua nappia toisen kerran näkömä näyttää listan kaikista suorittamattomista tapahtumista.

## 4.7 Tapahtumien suorittaminen

Tapahtuman suorittaminen aloitetaan *Tapahtumat*- tai *Harjoitteet*-näkömän kautta. Tapahtuma sisältää yhden tai useamman tehtävän, jonka kuntoutettavan toivotaan suorittavan. Jotkin tehtävätyyppien näytettävistä sisällöistä kaipaavat kuntoutettavan kosketussyötettä. Esimerkiksi monivalintatehtävässä kuntoutettavan on valittava vaihtoehdoista sopivin, ennen kuin hän voi edetä tapahtumassa eteenpäin.

Tapahtuman tehtäviä pääsee eteen- ja taaksepäin alareunan keski- ja oikealla painikkeella. Alareunan vasemmalla painikkeella kuntoutettava voi peruuttaa tapahtuman suorittamisen. Tällöin kuntoutettavalta varmistetaan vielä, että hän haluaa varmasti poistua tapahtumasta.

## 4.8 Harjoitteet-näkymä

Kuntoutettava pääsee *Harjoitteet*-näkyseen luvussa 4.3 esitetyn päänäkymän kautta. *Harjoitteet*- ja *Tapahtumat*-näkyvät ovat ulkoasultaan ja toiminnallisuuksiltaan lähes samanlaisia. *Harjoitteet*-näkyksessä näkyy kuntoutettavalle suunnattuja vapaaehtoisia tapahtumia, joita hän voi suorittaa useamman kerran tai niitä ei ole välttämättä suorittaa ollenkaan.

*Harjoitteet*-näkyksen vasemman alareunan painikkeella kuntoutettava voi palata takaisin luvun 4.3 näkyseen. Tapahtuman suorittamiseen siirrytään samalla tavalla kuin *Tapahtumat*-näkyksessä.

## 4.9 Jatkokehitykseen sovitut näkymät

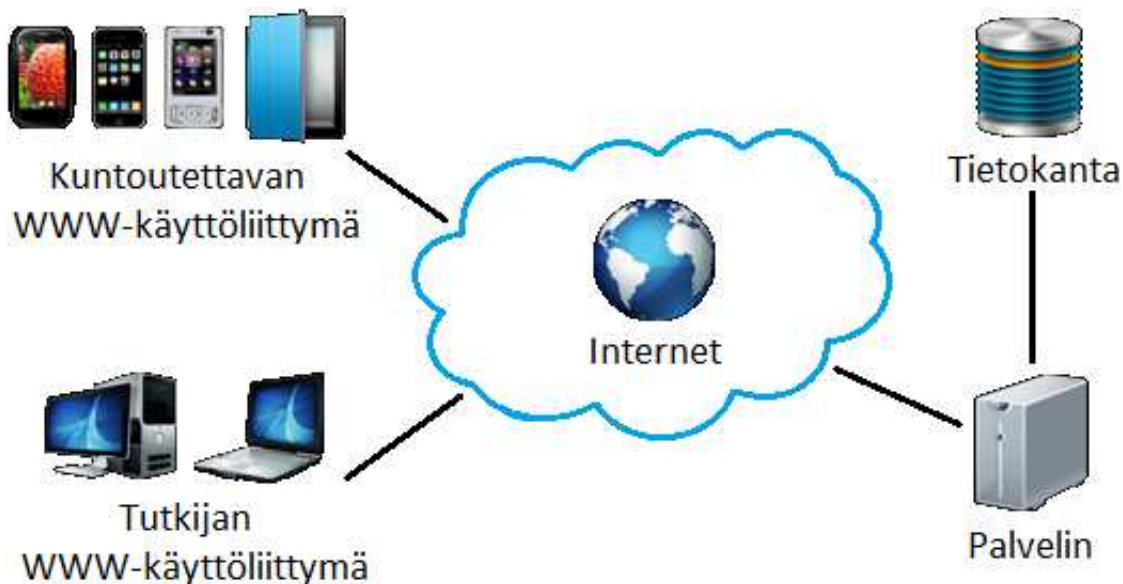
Tilaajan kanssa jatkokehitykseen sovitut näkymät ovat *Päiväkirja* sekä *Viestit*-näkyvät. *Päiväkirja*-näkyksessä kuntoutettava voi tallentaa päiväkirjamerkintöjä joko kuva-, ääni- tai tekstimuodossa. Kukin päiväkirjan merkintä tallentuu järjestelmän tietokantaan. *Viestit*-näkyksessä kuntoutettava voi tarkastella hänelle tulleita ryhmäkohtaisia viestejä, joita tutkija on lähettänyt oman käyttöliittymänsä kautta. *Viestit*-näkyksen olennaisin merkitys on toimia palautekanavana tutkijan ja kuntoutettavan välillä.

## 5 Tietojärjestelmän rakenne

Luvussa kuvataan sovelluksen rakenne ja sen toteutusratkaisut yleisellä tasolla. Sovellus koostuu kahdesta WWW-selainsovelluksesta, jotka ovat yhteydessä palvelimella olevaan projektissa suunniteltuun tietokantaan. Sovellus käyttää Vaadin-sovelluskehitysympäristön käyttöliittymäkirjastoja ja komponentteja. Prototyypin käyttöliittymistä kuntoutettavan mobiilisovellus noudattaa enemmän MVC-mallia, kun taas tutkijapuolen sovellus hyödyntää sitä vain osittain.

### 5.1 Tietojärjestelmän asiakas-palvelinarkkitehtuuri

Kuvassa 5.1 esitetään tietojärjestelmän yleinen rakenne sekä ulkopuoliset ohjelmit, joita järjestelmä käyttää. Palvelimella olevaan MySQL-tietokannanhallintajärjestelmään tallennetaan tietojärjestelmän tiedot. Tomcat-palvelin tarjoaa suorituslujan tutkijan ja kuntoutettavan käyttöliittymille, joita käytetään WWW-selainten kautta mobiililaitteilla ja tietokoneilla.

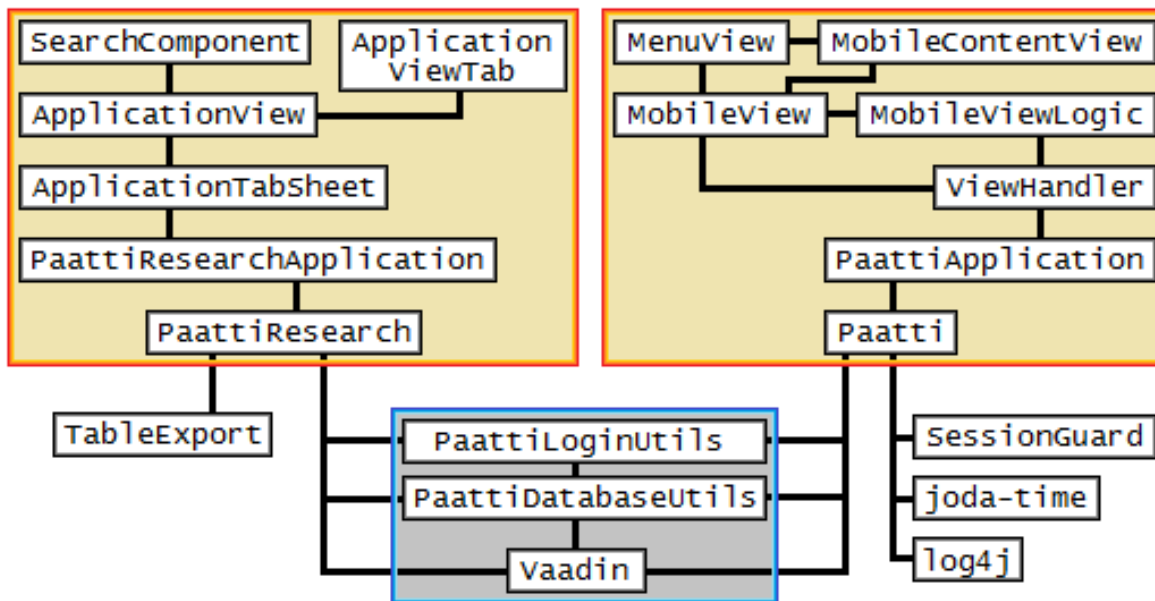


Kuva 5.1: Tietojärjestelmän asiakas-palvelinarkkitehtuuri.

Tietojärjestelmän toteutus aloitettiin maaliskuun alkupuolella kuntoutettavan käyttöliittymästä. Tietokannan suunnittelu ja tutkijan käyttöliittymän toteutus aloitettiin maaliskuun loppupuolella. Virtuaalipalvelimen ryhmä sai käyttöönsä huhtikuun alkupuolella testausta varten.

## 5.2 Tietojärjestelmän rakenne ja rajapinnat

Luvussa kuvataan tietojärjestelmään liittyvät ulkopuoliset komponentit lyhyesti. Tietojärjestelmä hyödyntää Vaadin-sovelluskehitysalustaa. Kuvassa 5.2 kuvataan projektissa toteutetun tietojärjestelmän arkkitehtuuri kerrosmallina.



Kuva 5.2: Tietojärjestelmän rakenne.

Tietojärjestelmä hyödyntää seuraavanlaisia ulkopuolisia komponentteja:

- **log4j** on Java-kirjasto lokien kirjoittamiseen,
- **joda-time** on Java-kirjasto, jota käytetään tapahtumien aikojen esittämisessä ja laskemisessa,
- **SessionGuard** on Vaadinin lisäosa, joka pitää palvelimen ja sovelluksen yhteyden päällä, jotta käyttäjän "istunto" ei vanhene sekä
- **TableExport** on Vaadinin lisäosa, jota käytetään suoritedatan tallentamiseen Excel-taulukkomuodossa.

## 5.3 Tietojärjestelmän komponentit

Luvussa kuvataan tiivistetysti sovelluksessa toteutettujen komponenttien käyttötarkoitusta. Kaikkien projektissa toteutettujen komponenttien tarkempi dokumentaatio löytyy projektin luokkadokumentaatiosta [?].

**PaattiLoginUtils**-paketti on kokoelma luokkia, joiden avulla sisäänkirjaututaan sovelluksiin. Komponentin lähdekoodi löytyy `PaattiLoginUtils`-kansioista.

**PaattiDatabaseUtils**-paketti on kokoelma luokkia, joiden avulla käytetään tietojärjestelmän tietokantaa. Komponentin lähdekoodi löytyy `PaattiDatabaseUtils`-kansioista.

**PaattiApplication** on kuntoutettavan käyttöliittymän ohjelma. Komponentin lähdekoodi löytyy `Paatti`-kansioista `paattiapplication`-paketista.

**MobileViewHandler**-komponentti on kuntoutettavan käyttöliittymän näkymähallitsija, joka hoitaa yhteyden kaikkien näkymien ja tietokannan välillä. Komponentin lähdekoodi löytyy `Paatti`-kansioista `paattiapplication`-paketista.

**MobileView**-komponentti on käyttöliittymän runko, johon asetetaan näkymän logiikka ja sisältö. Komponentin lähdekoodi löytyy `Paatti`-kansioista `mobileview`-paketista.

**MobileViewLogic**-komponentti on luokka, josta on peritty kuntoutettavan käyttöliittymän näkymien logiikat. Komponentin lähdekoodi löytyy `Paatti`-kansioista `mobileview`-paketista.

**MenuView**-komponentti on luokka, josta on peritty kuntoutettavan käyttöliittymän sovelluksen valikkonäkymät. Komponentin lähdekoodi löytyy `Paatti`-kansioista `menuview`-paketista.

**MobileContentView**-komponentti on luokka, josta on peritty kuntoutettavan käyttöliittymän sovelluksen näkymät. Komponentin lähdekoodi löytyy `Paatti`-kansioista `mobilecontentview`-paketista.

**PaattiResearchApplication**-komponentti on tutkijan käyttöliittymän ohjelma. Komponentin lähdekoodi löytyy `PaattiResearch`-kansioista `paattiapplication`-paketista.

**ApplicationTabSheet**-komponentti sisältää kaikki tutkijan käyttöliittymän sovelluksen näkymien välilehdet. Komponentin lähdekoodi löytyy `PaattiResearch`-kansioista `views`-paketista.

**ApplicationView**-komponentti on abstraktiluokka, josta on peritty tutkijan käyttöliittymän sovelluksen muut näkymät. Komponentin lähdekoodi löytyy `PaattiResearch`-kansioista `views`-paketista.

**ApplicationViewTab**-komponentti on abstraktiluokka, josta on peritty tutkijan käyttöliittymän sovelluksen näkymissä olevat välilehdet. Komponentin lähdekoodi löytyy PaattiResearch-kansiosta views-paketista.

**SearchComponent**-komponentti on tutkijan käyttöliittymän näkymissä oleva haku-komponentti. Komponentin lähdekoodi löytyy PaattiResearch-kansiosta components-paketista.

## 5.4 Toteutusratkaisujen kehittyminen

Kuntoutettavan käyttöliittymän toteutusratkaisujen suurin muutos tapahtui näkymien näyttämisessä. Kuntoutettavan käyttöliittymä latsi sovellusta käytettäessä seuraavan näkymän uudelleen aina kun näkymästä siirryttiin toiseen. Myöhemmin kuntoutettavan käyttöliittymän näkymien latausta muutettiin niin, että kaikki sovelluksen näkymät luodaan vain kerran silloin, kun niihin mennään sovelluksessa ensimmäisen kerran. Sovellus tallettaa luodut näkymät ja lataa näytettäviä näkymiä uudelleen sisäisestä tietorakenteesta.

Tutkijan käyttöliittymän toteutusratkaisujen suurin muutos tapahtui projektin viimeistelyvaiheessa toukokuun alkupuolella. Sovelluksen rakennetta muutettiin enemmän noudattamaan MVC-arkkitehtuuria. Sovellusta myös refaktoroitiin hyödyntämään enemmän perintää ja rajapintoja koodin siistimisen ja sovelluksen suorituskyvyn parantamisen vuoksi.



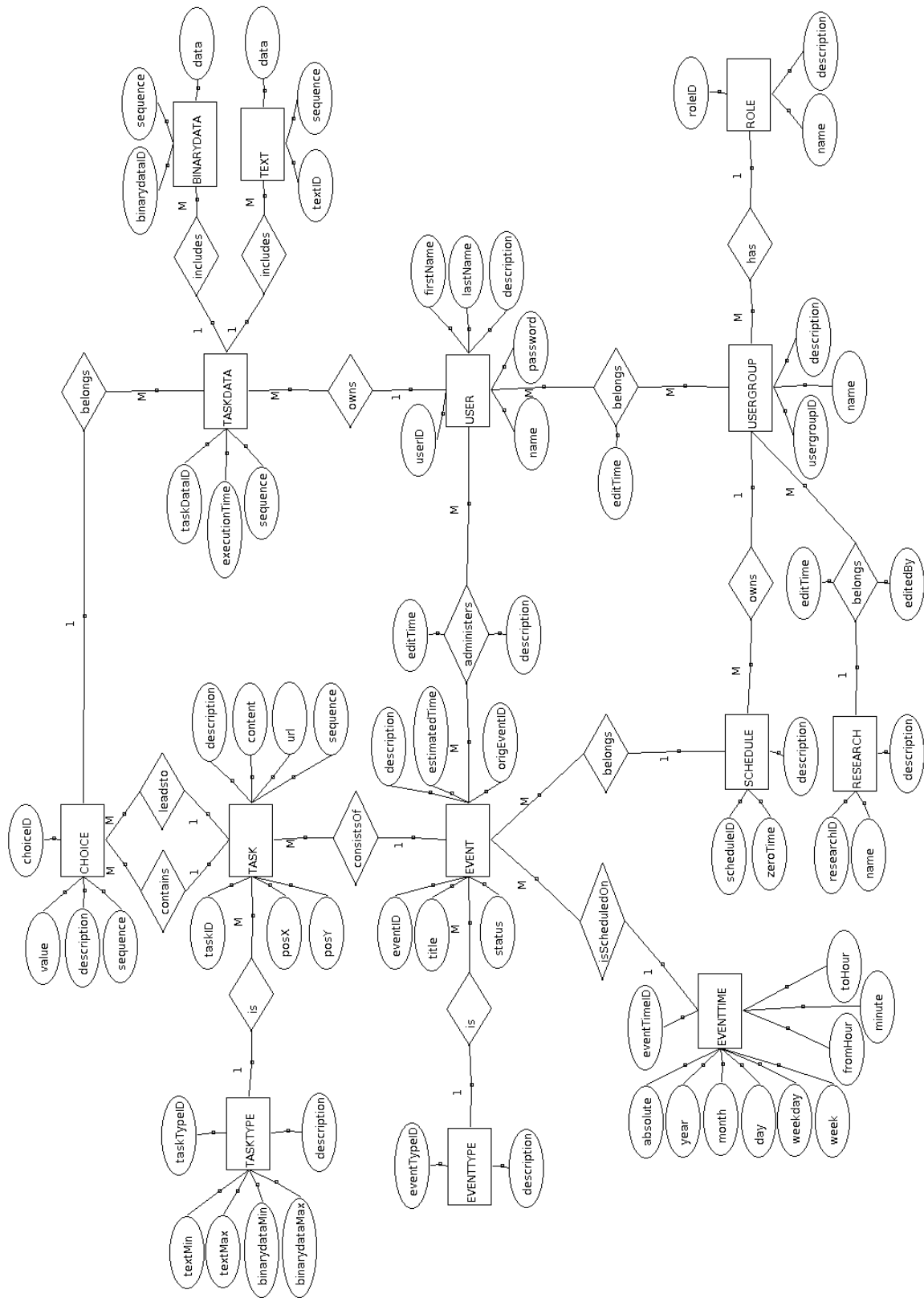
## 6 Tietojärjestelmän tietokanta

Projektryhmä suunnitteli tietokannan järjestelmälle. Tietokantaan tallennetaan tutkijan luomia käyttäjiä, ryhmiä, tutkimuksia, kuntoutusohjelmia, tapahtumia ja kuntoutettavan tapahtumien suorittamisesta tulevaa suoritodataa.

### 6.1 Tietokannan ER-kaavio

Kuvan 6.1 ER-kaaviossa on esitetty tietokannan taulut ja niiden kentät sekä suhteet toisiin tietokannan tauluihin. Luvuissa 6.2 - 6.15 kuvataan tietokannan taulujen ja kenttien merkitys tarkemmin sekä kuvataan esimerkkitilanteita taulujen käyttöön liittyen.

Kuvan 6.1 tauluihin ei ole piirretty `rowStatus`-kenttää, sillä se on jokaisella taululla kenttänä. `RowStatus`-kenttään tallennetaan taulukossa olevan sarakkeen tila (kuten `ACTIVE` tai `DELETED`).



Kuva 6.1: Järjestelmän tietokanta ER-kaaviona.

## 6.2 USER-taulu

USER-tauluun tallennetaan järjestelmän käyttäjät. Taulu sisältää seuraavanlaiset kentät:

- `userID` on käyttäjän tietokannan id-numero,
- `firstName` on käyttäjän etunimi,
- `lastName` on käyttäjän sukunimi,
- `name` on käyttäjän käyttäjätunnus,
- `password` on käyttäjän salasana sekä
- `description` on lyhyt kuvaus käyttäjästä.

USER-taulu on kytköksissä TASKDATA, EVENT ja USERGROUP-tauluihin.

## 6.3 USERGROUP-taulu

USERGROUP-tauluun tallennetaan järjestelmän ryhmät. Taulu sisältää seuraavanlaiset kentät:

- `groupID` on käyttäjän tietokannan tunnistenumero,
- `name` on ryhmän nimi sekä
- `description` on lyhyt kuvaus ryhmästä.

## 6.4 ROLE-taulu

ROLE-tauluun tallennetaan järjestelmän ryhmien roolityypit. Taulu sisältää seuraavanlaiset kentät:

- `roleID` on roolin tietokannan tunnistenumero sekä
- `name` on roolin nimi.
- `description` on lyhyt kuvaus roolista.

## 6.5 RESEARCH-taulu

RESEARCH-tauluun tallennetaan järjestelmän tutkimukset. Taulu sisältää seuraavanlaiset kentät:

- `researchID` on tutkimuksen tietokannan tunnistenumero,
- `name` on tutkimuksen nimi sekä
- `description` on lyhyt kuvaus tutkimuksesta.

## 6.6 SCHEDULE-taulu

SCHEDULE-tauluun tallennetaan järjestelmän kuntoutusohjelmat. Taulu sisältää seuraavanlaiset kentät:

- `scheduleID` on kuntoutusohjelman tietokannan tunnistenumero,
- `zeroTime` on kuntoutusohjelman tallennusajankohta sekä
- `description` on lyhyt kuvaus kuntoutusohjelmasta.

## 6.7 EVENT-taulu

EVENT-tauluun tallennetaan järjestelmän tapahtumat. Taulu sisältää seuraavanlaiset kentät:

- `eventID` on tapahtuman tietokannan tunnistenumero,
- `title` on tapahtuman otsikko,
- `status` on tapahtuman tila,
- `origEventID` on tapahtuman alkuperäisyyden ilmaisemiseen käytettävä tunnistenumero,
- `estimatedTime` on tapahtuman arvioitu kesto sekä
- `description` on lyhyt kuvaus tapahtumasta.

## 6.8 EVENTTIME-taulu

EVENTTIME-tauluun tallennetaan aikataulutettujen tapahtumien ajoitus kuntoutusohjelmassa. Taulu sisältää seuraavanlaiset kentät:

- `eventTimeID` on ajan tietokannan tunnistenumero,
- `absolute` ilmaisee ajan absoluuttisuuden,
- `year` ilmaisee ajan vuotta,
- `month` ilmaisee ajan kuukautta,

- `day` ilmaisee ajan päivää,
- `weekday` on ajan viikonpäivä,
- `week` ilmaisee ajan viikkoa,
- `fromHour` on ajan aloituskohta,
- `toHour` on ajan lopetuskohta sekä
- `minute` ilmaisee ajan minuuttia.

Kentät `fromHour` ja `toHour` ilmaisevat ajanjaksoa, jolloin tapahtuma on tehtävissä. Jos kuntoutettava ei suorita tapahtumaa asetetulla ajankohdalla, tapahtuma siirtyy rästitapahtumaksi.

## 6.9 EVENTTYPE-taulu

EVENTTYPE-tauluun tallennetaan tapahtumatyypit (kuten kysely ja liikunta). Taulu sisältää seuraavanlaiset kentät:

- `eventTypeID` on tapahtumatyyppin tietokannan tunnistenumero sekä
- `description` on tapahtumatyyppin kuvaus.

## 6.10 TASK-taulu

TASK-tauluun tallennetaan tapahtumiin liittyvät tehtävät. Taulu sisältää seuraavanlaiset kentät:

- `taskID` on tehtävän tietokannan tunnistenumero,
- `content` on tehtävän otsikko,
- `url` on tehtävän tila,
- `sequence` on tehtävän järjestys tapahtumassa,
- `description` on lyhyt kuvaus tehtävästä,
- `posX` on tapahtuman x-koordinaatti piirtoalueella sekä
- `posY` on tapahtuman y-koordinaatti piirtoalueella.

## 6.11 TASKTYPE-taulu

TASKTYPE-tauluun tallennetaan tehtävätyypit (kuten teksti ja monivalinta). Taulu sisältää seuraavanlaiset kentät:

- `taskTypeID` on tehtävätyypin tietokannan tunnistenumero,
- `textMin` on tehtävätyypin tekstimuotoisen datan minimi,
- `textMax` on tehtävätyypin tekstimuotoisen datan maksimi,
- `binaryMin` on tehtävätyypin binäärimuotoisen datan minimi,
- `binaryMax` on tehtävätyypin binäärimuotoisen datan maksimi sekä
- `description` on lyhyt kuvaus tehtävätyypistä.

## 6.12 CHOICE-taulu

CHOICE-tauluun tallennetaan tehtävien sisällä olevat valinnat. Taulu sisältää seuraavanlaiset kentät:

- `choiceID` on valinnan tietokannan tunnistenumero,
- `value` on valinnan arvo,
- `sequence` on valinnan järjestys tehtävässä sekä
- `description` on lyhyt kuvaus valinnasta.

## 6.13 TASKDATA-taulu

TASKDATA-tauluun tallennetaan tapahtumien suoritedataan liittyvät tiedot. Taulu sisältää seuraavanlaiset kentät:

- `taskdataID` on suoritedatan tunnistenumero,
- `executionTime` on suoritedatan suoritusajankohta sekä
- `sequence` on suoritedatan järjestys tehtävässä.

## 6.14 BINARYDATA-taulu

BINARYDATA-tauluun tallennetaan binäärimuotoinen suoritedata (kuten kuva ja ääni). Taulu sisältää seuraavanlaiset kentät:

- `binarydataID` on binäärimuotoista dataa sisältävän suoritedatan tunnistenumero sekä
- `sequence` on datan järjestys suoritedatassa.

## 6.15 TEXTDATA-taulu

TEXTDATA-tauluun tallennetaan tekstimuotoinen suoritedata. Taulu sisältää seuraavanlaiset kentät:

- `textID` on tekstimuotoista dataa sisältävän suoritedatan tunnistenumero sekä
- `sequence` on datan järjestys suoritedatassa.

## 7 Ohjelmointikäytännöt

Luvussa kuvataan projektin ohjelmointikäytännöt. Projektissa noudatettiin suunnitteluvaiheessa määriteltyjä ohjelmointikäytännöitä.

### 7.1 Lähdekoodin muotoilu ja tiedostojen nimeäminen

Lähdekoodi ohjelmoitiin projektisuunnitelmassa [5] sovittujen käytänteiden mukaisesti käyttäen mallina Javan yleisiä käytänteitä [2]. Luokkien nimissä ensimmäinen kirjain kirjoitettiin pienellä ja sanat kirjoitettiin yhteen isoin alkukirjaimin. Funktioiden ja muuttujien nimissä ensimmäinen kirjain kirjoitettiin pienellä ja sanat kirjoitettiin yhteen isoin alkukirjaimin. Lähdekooditiedostojen hakemistorakenne oli valmiiksi määritelty Vaadin-sovelluskehitysympäristössä, eikä rakennetta muutettu. Muiden tiedostojen ja hakemistojen nimet kirjoitettiin pienillä kirjaimilla ja englanniksi.

Tiedostojen ja lähdekoodin nimeäminen toteutui suunnitelman mukaisesti ja verkkosivulle ja CD:lle sijoitetut dokumentit vastasivat edellämainittuja käytänteitä.

### 7.2 Lähdekoodin kommentointi

Lähdekoodit kommentointiin englanniksi. Kommentit aloitettiin yksittäisten rivien tai rivin loppujen tapauksissa merkillä `//`. Useamman rivin tapauksessa kommentti aloitettiin merkillä `/*` ja päätettiin merkillä `*/`. Luokkien ja funktioiden kommentit aloitettiin `/**` ja päätettiin `*/`. Luokkadokumentointi muodostettiin JavaDocin avulla, jonka kommentointiohjeita noudatettiin lähdekoodissa.

### 7.3 Lähdekoodiesimerkki

Sovelluksessa toteutettua lähdekoodin sisennystä ja kommentointia havainnollistaa seuraava esimerkki:



```
/**
 * Layout for all the mobile views has slots for header,
 * footer and the content.
 *
 * @author Lauri Satokangas, lauri.n.satokangas@student.jyu.fi
 * @date 13.5.2012
 */
public abstract class MobileView extends VerticalLayout {

    private Panel pageContent;
    private final HashMap<Object, Component> footerComponents;

    // This is the left button in the footer layout.
    public Object FOOTER_BUTTON_LEFT = "leftfooterbutton";

    /**
     * Get the component in the footer.
     * @return footerComponent
     */
    public HashMap<Object, Component> getFooterComponents() {
        return footerComponents;
    }
}

...

```

## 7.4 Kehitysympäristö

Sovelluksen kehitystyökaluna käytettiin NetBeans IDE:n versiota 7.1. Lisäosina käytettiin Maven-automatisointityökalua lähdekoodin kääntämiseen ja Vaadin-sovelluskehitysalustan käyttöliittymäkirjastoja ja komponentteja sovelluksen toteutukseen.

## 7.5 Testauksen käytänteet

Sovellusten toteutusvaiheessa jokainen ryhmän jäsen toteutti käytännön yksikkötestausta heti koodiin tekemiensä muutoksen jälkeen omatoimisesti. Ajan puutteen

vuoksi tietojärjestelmälle ei laadittu järjestelmätestaussuunnitelmaa eikä ryhmä myöskään ehtinyt luoda yksikkötestejä sovelluksille. Muutama testitiedosto luotiin tietokantaa käsitteleville tiedostoille. Integraatiotestausta ryhmä suoritti projektin viimeistelyvaiheessa jonkin verran.

Ryhmälle jaettiin ajoissa ennen toteutuksen alussa erilaisia mobiililaitteita ja sormitietokoneita. Niiden avulla ryhmä testasi kuntoutettavan sovellusta tietyn väliajoin ja sai näin arvokasta tietoa liittyen sovelluksen toimivuuteen eri mobiililaitteissa.

Ryhmälle järjestettiin maaliskuun puolessavälissä käytettävyysspäivä Meeri Mäntylän toimesta. Käytettävyysspäivän aikana ryhmä sai käyttöliittymien käytettävyyteen liittyvää tietoa ja palautetta projektissa toteutettavista käyttöliittymistä [käytettävyysspäivämuistio lähteeksi].

Sovelluksella oli vain muutama ulkopuolinen testaja projektin aikana. Laajimman palautteen sovellusten toimivuudesta projektiryhmä sai vastaavalta ohjaajalta Jukka-Pekka Santaselta [useita sähköposteja]. Sovelluksen testaukseen osallistui myös projektin jatkokehityksestä kiinnostunut Tero Kallioinen [pari sähköpostia]. Projektin tilaajien kiireellisen aikataulun vuoksi tilaajilta saatiin vain vähän palautetta liittyen sovellusten käyttöliittymien toimivuuteen. Tilaajien ainoat käyttöttestaukset tapahtuivat projektin kahdessa viimeisessä kokouksessa. Projektin tilaajilta saatiin kuitenkin testidataa testiaineiston luomiselle ja sitä hyödynnettiin projektissa jonkin verran.

Ryhmän jäsenet harjoittivat myös omatoimista käyttäjätestausta havaitakseen sovellusten toiminnallisuuksiin liittyviä puutteita ja bugeja. Ryhmä testasi sovellusten toiminnallisuuksien toimivuutta tasaisin väliajoin, varsinkin ennen projektin loppuesitystä.

Projektiryhmä ei ajan puutteen vuoksi ehtinyt testaamaan sovelluksia niiden loppukäyttäjillä. Tämä asia on otettava huomioon jatkokehityksessä sekä järjestettävä kuntoon mahdollisesti jatkokehityksen alkuvaiheessa.

## 7.6 Testauksen tulokset

Ryhmä sovellusten testausten tulokset osoittivat, että toteutetut sovellukset toimivat asetettujen pakollisten ja tärkeiden toiminnallisten vaatimusten mukaisesti. Usein toistuvia virheitä ei löydetty testausten aikana. Muita sovellusten tunnettuja puutteita ja virheitä kuvataan tarkemmin luvussa 8.2.

## 8 Tavoitteiden toteutuminen

Luvussa kuvataan projektin vaatimusten toteutuminen, tunnetut virheet ja puutteelliset toteutusratkaisut sekä jatkokehitysideat. Projektissa saatiin toteutettua 42 42:sta pakollisesta tai tärkeästä ja 24 88:sta mahdollisesta tai jatkokehitykseen asetetusta toiminnallisesta vaatimuksesta. Tilaajan kanssa sovittiin osia toteutuksesta jo aikaisessa vaiheessa jatkokehitykseen rajallisen ajan vuoksi. Jatkokehitykseen sovittiin tutkijan käyttöliittymässä mediatiedostojen hallintanäkymä, suoritiedon raporttinäkymä, viestien lähetys- ja lukunäkymä sekä ohjenäkymä. Kuntoutettavan käyttöliittymästä jatkokehitykseen sovittiin kiihtyvyyssanturidatan lukeminen, päiväkirjamerkintöjen luku- ja talletusnäkyvä ja viestien lähetys- ja lukunäkymä.

### 8.1 Vaatimusten toteutuminen

Luvussa kuvataan pääpiirteittäin sovellukselle asetettujen vaatimusten toteutuminen. Vaatimusmäärittely [1] käsittelee tarkemmin vaatimuksia ja niiden toteutumista.

**Tutkijan käyttöliittymän 63:sta vaatimuksesta** toteutettiin kaikki 24 pakollisiksi tai tärkeiksi asetettua vaatimusta. *Oma sivu, Media, Raportti* ja *Ohje*-näkyviä ei toteutettu projektissa ja ne siirrettiin jatkokehitykseen.

**Kuntoutettavan käyttöliittymän 48:sta vaatimuksesta** toteutettiin kaikki 11 pakollisiksi tai tärkeiksi asetettua vaatimusta. Vaatimus yksinkertaisten kuvakkeiden lisäämisestä näkyviin vieviin painikkeisiin ja tapahtuman tyyppien ilmaisuun sekä *Päiväkirja, Viestit* ja *Ohje*-näkyviin liittyvät vaatimukset sovittiin jatkokehitykseen.

**Tapahtuman suorittamisen 8:sta vaatimuksesta** toteutettiin kaikki 3 pakollisiksi tai tärkeiksi asetettua vaatimusta. Kuntoutettava voi suorittaa tapahtumia ja keskeyttää suoritettavan tapahtuman.

**Sisäänkirjautumiseen liittyvistä 3:sta mahdollisesta vaatimuksesta** toteutettiin 2 projektissa. Sisäänkirjautuminen tallentaa kirjautumistiedot keksiin, jolloin esimerkiksi kuntoutettavan tarvitsee kirjautua sovellukseen vain yhden kerran, ellei hän poista keksitietoja oman mobiililaitteen selaimesta. Kirjautumistietojen salaamiseen liittyvä vaatimus sovittiin jatkokehitykseen.

**Tapahtumien tehtäväkomponentteihin liittyvistä 19:sta vaatimuksesta** toteutet-

tiin kaikki 7 pakolliseksi tai tärkeiksi asetettua vaatimusta. Media-, sensori-, syöte-, liikusäädin- ja hälytystehtäväkomponentteja ei toteutettu projektissa, ja ne sovittiin jatkokehitykseen.

**Suoritedatan tallentamiseen eri tiedostomuodoissa** projektiryhmä toteutti vain xls-muotoiseen dataan liittyvän vaatimuksen. Muissa tiedostomuodoissa vietyä dataa ei ehditty projektin aikana edes harkitsemaan.

**Kiihtyvyyssanturidatan ja muiden sensoridatojen keräämiseen asetettuja vaatimuksia** ei ehditty toteuttaa projektin aikana. Kuntoutettavan käyttöliittymää ei asetettu projektissa hyödyntämään PhoneGap-rajapintaa. Kyseiset vaatimukset sovittiin tilaajan kanssa jatkokehitykseen.

**Viesteihin liittyviä vaatimuksia** ei ehditty toteuttaa projektissa. Ryhmäkohtaisilla viesteillä tutkija voisi tiedottaa haluamiaan ryhmiä jostain tarpeellisista asioista. Viestien tarkoitus on luoda eri käyttäjäkuntien välille vuorovaikutus- ja palautekanavia.

Projektissa toteutettuja vaatimuksia ei kirjattu testatuiksi tai hyväksytyiksi, koska suunnitelmallista ja hallittua järjestelmätestausta ei ehditty suorittaa projektin aikana.

## 8.2 Tunnetut puutteet ja virheet

Tutkijan sovelluksessa kuntoutusohjelmien aikatauluttamisnäkyssä **tutkijan ei ole mahdollista valita kaikkia tietokannan mahdollistavia ajoitukseen liittyviä ominaisuuksia**. Tällaisia ovat mm. absoluuttisen ajan asettaminen ja aikaväli, jolloin tapahtuman voi suorittaa.

Tutkijan sovelluksessa kuntoutusohjelmien **aikatauluttamisessa käytettävään ajanasetuskomponenttiin ei ole ohjelmoitu tarkistinta**, jos tutkija syöttää syöteriville virheellisen aikatiedon. Ongelmaa ei tule, jos tutkija asettaa ajan vieressä olevan painikkeen painalluksen kautta avautuvasta päivämäärän ja ajan asetusikkunasta.

**Uuden kuntoutusohjelman tallentaminen luo tapahtumista uudet kopiot tietokantaan** suoritedatan tunnistamista varten. Näin käy myös, jos tutkija muokkaa ja tallentaa kussakin välissä jo olemassa olevan kuntoutusohjelman. Jos esimerkiksi tutkija luo kymmenen tapahtumaa sisältävän kuntoutusohjelman, yhdellä tapahtumalla on kymmenen monivalintatehtävää ja jokaisella niistä on kymmenen va-

lintaa, tulee uuden kuntoutusohjelman tallennuksessa luotua kokonaisuudessaan 1001 uutta taulualkiota tietokantaan. Tästä kopioiminen voi aiheuttaa ongelmia jo pelkästään yhden käyttäjän toimesta.

Kuntoutusohjelmaa tallentaessa heittää virheen, jos kuntoutusohjelmaan ei ole lisätty tapahtumia.

**Järjestelmän käyttäjien roolit määräytyvät käyttäjän ryhmien jäsenyyksien mukaan.** Tämä tarkoittaa sitä, että yksi käyttäjä voi olla eri rooleissa eri ryhmissä. Järjestelmän rooleja on ajateltu käytettävän sovellusten käyttöoikeuksien jakamiseen. Esimerkiksi tutkijalla voisi olla täydet käyttöoikeudet tutkijan sovelluksessa, sekä omaisella tai opiskelijalla voisi olla rajoitetut tutkijan oikeudet sovelluksen käyttöön. Käyttöoikeuksien määräämisessä voi syntyä ongelmia silloin, kun yksi käyttäjä on jäsenenä useassa ryhmässä eri rooleilla. Ongelman voisi ratkaista asettamalla käyttäjäroolit USER ja USERGROUP-taulujen välisen BELONGS-taulun kautta.

**Kuntoutettavan sovelluksen tapahtuman suorittamisessa on pieni bugi.** Kun kuntoutettava on monivalintatehtävissä, valitsee hän yhden vaihtoehdon ja menee tapahtumassa eteenpäin. Jos kuntoutettava palaa takaisin edelliseen tehtävään ja ei valitse vaihtoehtoa ja yrittää mennä tapahtumassa eteenpäin, niin tapahtuman teko päättyy siihen.

Käyttäjien salasanoihin liittyy pari puutteellista ratkaisua. **Käyttäjien salasanvoja ei tallennetta salatussa muodossa tietokantaan** ja tutkijan käyttöliittymässä **tutkija näkee tarkasteltavan käyttäjän salasanan *Käyttäjän tiedot* -alivälilehdellä.** Nämä asia tulisi ottaa käsittelyyn projektin jatkokehityksen alkuvaiheessa.

**Tutkija ei pysty tällä hetkellä hakemaan käyttäjiä kuin käyttäjän käyttäjätunnuksen mukaan.** Tutkijan olisi tarkoitus halutessaan myös voitava hakea tietokannasta käyttäjiä sekä etunimen tai sukunimen mukaan.

**Näkymien hakukomponentin *Hae*-painikkeella pystyy myös valitsemaan valitun kohteen tiedot tarkasteltavaksi *Valitse*-napin tapaan.** Hakukomponentin *Hae*-painikkeella olisi vain tarkoitus hakea hakukenttään syötetyn tekstin perusteella tietoa tietokannasta.

**Tietokannan BELONGS-taulu käyttää avaimenaan muokkausaikaa koosteavaimen sijaan.** Muokkaus aika toimii tällä hetkellä vertailukohteena, kun tutkija poistaa ja tai päivittää tarkasteltavaa kohdetta. Koosteavain koostuisi käyttäjän ja käyttäjän ryhmän ID-numerosta.

**Tapahtumamuokkaimen alueen piirto tapahtuu palvelimella ja palvelin lähettää aina uudelleenpiirretyyn kuvan asiakaspääteelle.** Tapahtumanmuokkaimen alueen piirron voisi asettaa jatkokehityksessä tapahtumaan asiakaspäässä, jos se on mahdollista. Ongelmia voi syntyä tapahtumamuokkaimen käytettävyyteen silloin, jos tapahtumamuokkainta muutetaan niin, että piirtoalueesta lähetettävien kuvien koko kasvaa suureksi.

Tutkijan käyttöliittymässä **tutkijalta ei kysytä varmennusta hänen tallentaessa muutoksia tai poistaessa valittua kohdetta.** Tällä voisi estää tahattomat poistot ja muutosten tallennukset. Vaihtoehtoisesti tutkija voisi asetuksistaan ottaa pois päältä varmennuksen kysymisen, jos jatkuva varmentaminen haittaa käyttöliittymän käytettävyyttä.

### 8.3 Jatkokehitysideat

Tutkijan käyttöliittymän yläreunaan päävälilehden yläpuolelle tulisi asettaa kapea alue, jossa näkyisi sovelluksen nimi ja jossa olisi oikeassa yläreunassa uloskirjautumispainike. Tällöin sovelluksesta uloskirjautuminen voisi tapahtua mistä välilehden näkymästä tahansa.

Tutkijan käyttöliittymä kaipaa vihjetekstejä useisiin paikkoihin. Ulkopuolisten käyttötötestauksen palautteessa ilmeni, että tapahtumamuokkaimen käyttämisen oppiminen ei välttämättä ole aloittelijalle kaikista helpoin tehtävä.

Kiihtyvyyssanturidatan keräämiseen käytettävälle PhoneGap-rajapinnalle on mahdollisesti tulossa tuki Vaadin-sovelluskehitysalustaan tulevaisuudessa. Jos kyseinen laajennus tulee saataville lähitulevaisuudessa, niin jatkokehitysryhmän kannattaa odottaa kyseistä laajennusta ennen kiihtyvyyssanturiin liittyvien toiminnallisten vaatimusten toteuttamista.

Käyttäjän syöttämiä syötteitä syötekenttiin ei tarkasteta millään lailla. Syötteiden tarkistuksella voitaisiin eliminoida `DROP TABLE * -`tyyppisten syötteiden meno tietokantaan.

*Oma sivu* -näkyvä tulisi vaihtaa *Viestit* -näkymäksi. Siinä tutkija voisi tarkastella kuntoutettavilta tulleita viestejä ja hälytyksiä sekä lähettää ryhmille ryhmäkohtaisia viestejä. Kyseisen näkymän tulisi myös olla päänäkyvä, kun tutkija kirjautuu sisään tutkijan käyttöliittymään.

Tietojärjestelmälle voisi asettaa oman testitietokannan tietokannan testaukselle ja testaustiedostoille. Näin tietokantaa voisi testata eri kannalla, kuin mitä tietojärjestelmä käyttäisi.

Tiedon salauksen tietokantaan tallentaessa voisi asettaa tapahtumaan muillekin kuin salasanoille. Tällaisia kohteita voisi olla käyttäjien etunimi, sukunimi ja käyttäjätunnus. Tällä hetkellä tietokantaan tallentuu vain salaamatonta dataa.

Lokituksen voisi asettaa tapahtumaan sovelluksia käytettäessä. Lokitus helpottaisi järjestelmän virheiden havaitsemista ja kyseiset tiedostot voi helposti arkistoida muualle.

Tapahtuman muokkauksesta syntyvää muutoshistoriaa tulisi pystyä tarkastelemaan tapahtumaa muokatessa. Muutoshistorian voisi sijoittaa omaan alivälilehteen, josta tutkija voisi tarkastella tapahtumaan tehtyjä muutoksia.

Välilehtien sisälle sijoitetut *Tallenna* ja *Poista*-painikkeet voisivat sijaita välilehtien ulkopuolella alhaalla, jolloin tallentamisen ja poistamisen voisi tehdä riippumatta päällä olevasta välilehdestä.

Toteutettujen käyttöliittymien ulkoasuteemoihin ei ehditty käyttämään tarpeeksi aikaa projektin aikana. Käyttöliittymien tema-asetuksia voi muokata CSS-tyylitiedostojen kautta. Käyttöliittymien graafista linjaa kannattaa kysyä paremmin asiaan perehtyneiltä.

Kaikkien toimintojen järjestystä käyttömäärien ja yleisyyden mukaan ei ehditty projektin aikana käyttämään tarpeeksi aikaa. Esimerkiksi hakukomponentin toiminnot voisi sijoittaa yleisimmästä harvemmin käytettyihin painikkeisiin ylhäältä alas.

Tutkijan käyttöliittymässä tutkijoilla voisi olla tapahtumien näkemiseen ja muokkaamiseen liittyviä erityisoikeuksia. Esimerkiksi tapahtuman näkeminen hakulistalla ja muokkaus voisi olla rajattu vain sen luoneelle ja muokanneille sekä tapahtumaan sidotun ryhmän tutkijoille. Tietokannan taulujen `rowStatus`-kentän toimintaa voisi laajentaa kattamaan kyseisen ominaisuuden käyttöliittymässä.

Tapahtumamuokkaimeen liittyy useita parannusehdotuksia. Yhden tehtävän siirtäminen piirtoalueella tulisi olla mahdollista raahamalla tapahtumaa hiiren painike pohjassa. Tehtävien etenemissuunnan tehtävien välillä tulisi esittää joko nuolella tai vastaavalla havainnollistavalla visualisoinnilla. Piirtoalueella näytettävät tehtävätyypit tulisi visuaalisesti olla erotettavissa muun kuin värin avulla. Tapahtumamuokkaimessa tulisi näyttää tapahtumassa olevat tehtävät listana käyttäjälle. Tutki-

jalla tulisi lisäksi myös olla mahdollisuus tehdä uusia tapahtumatyyppejä tapahtumille.

Tutkijan tulisi olla mahdollista tehdä jaksottain tapahtuvia kuntoutusohjelmia. Tämä tarkoittaisi sitä, että tutkijan olisi mahdollisuus tehdä tapahtumalle aika, joka toistuu tasaisesti tietyssä aikana. Esimerkiksi tutkija voisi asettaa tapahtuman suoritettavaksi kerran viikossa yhden kuukauden ajan.



## 9 Käyttöohjeita ylläpitäjälle

Luvussa kuvataan lyhyesti, kuinka järjestelmän voi asettaa kuntoon pyörimään palvelimelle, kuinka tutkijan käyttöliittymään pystyy lisäämään uuden näkymän ja kuinka tapahtumamuokkaimissa käytettävien tehtäväkomponenttien lisäys tapahtuu.

### 9.1 Sovelluksen asennus palvelimelle

TODO

### 9.2 Tietokokonaisuuden hallintanäkymän lisääminen

TODO

### 9.3 Tehtäväkomponenttien lisääminen

TODO

## 10 Yhteenveto

Paatti-projekti toteutti Sovellusprojekti-kurssilla keväällä 2012 Jyväskylän yliopiston psykologian laitokselle prototyypin tietojärjestelmästä, jolla tuetaan muun muassa iäkkäiden, työuupumuksesta kärsivien ja mielenterveyspotilaiden itsenäistä elämää ja kuntoutumista sekä tuetaan tutkijoiden tekemää tutkimusta.

Projektissa toteutettiin prototyypit vaadituista kahdesta sovelluksesta, joista toista käytetään kuntoutusohjelmien luomiseen ja niiden aikatauluttamisen ryhmille sekä toista kuntoutusohjelmien seuraamiseen ja tekemiseen. Toteutettu järjestelmäkokonaisuus toimii asetettujen päätoiminnallisuuksien mukaisesti ja antaa siten hyvän pohjan tuotantoversion kehittämiseksi.

Tietojärjestelmässä tutkija luo tutkimuksiinsa liittyen kuntoutettavista ryhmiä, joille määritetään omat kuntoutusohjelmat. Kuntoutusohjelmat sisältävät aikataulutettuja tapahtumia, jotka sisältävät määritetyssä järjestyksessä suoritettavia tehtäviä. Tehtävät voivat olla kysymyksen tai ohjeen muodossa ja voivat sisältää tekstiä tai monivalinnan.

Kuntoutettavia voi siirtää ryhmästä toiseen edistymisen mukaan tutkijan toimesta. Kuntoutettava voi olla monessa ryhmässä, jolloin hän suorittaa useampaa kuntoutusohjelmaa samanaikaisesti. Tutkija pystyy seuraamaan kuntoutettavan tai ryhmän edistymistä omasta käyttöliittymästään. Tietojärjestelmän tietokantaan keräämää tietoa saa myös siirrettyä toisiin järjestelmiin xls-tiedostomuodossa.

Kuntoutettava näkee suoritettavat kuntoutusohjelman tapahtumat mobiililaitteessa olevasta WWW-sovelluksesta. Tapahtumat voivat olla esimerkiksi liikuntaharjoitteita, rentoutusharjoitteita tai lääkkeiden ottamisen muistutuksia.

Projektin rajallinen aikataulu ei mahdollistanut kaikkien tilaajien vaatimusten toteuttamista ja siksi toteutettu järjestelmäprototyyppi kaipaa jatkokehitystä. Toteutunut järjestelmäprototyyppi sovittiin tilaajien kanssa vietäväksi jatkokehitykseen, jossa kehitetään projektin aikana toteuttamatta jääneet toiminnallisuudet sekä korjataan sovelluksissa esiintyvät puutteet ja ongelmat.

## 11 Lähteet

- [1] Keränen Tapio, Salminen Toni, Salokangas Jari ja Satokangas Lauri, "Paatti-projekti, Vaatimusmäärittely", Jyväskylän yliopisto, tietotekniikan laitos, 2012.
- [2] Oracle, "Code Conventions for the Java Programming Language", saatavilla PDF-muodossa <URL: <http://www.oracle.com/technetwork/java/codeconventions-150003.pdf>>, viitattu 25.3.2012.
- [3] Santanen Jukka-Pekka, "Tietotekniikan Sovellusprojektien ohje", saatavilla HTML-muodossa <URL: <http://www.mit.jyu.fi/opetus/sovellusprojektit/projohje.html>>, Jyväskylän yliopisto, tietotekniikan laitos, 11.9.2006.
- [4] Keränen Tapio, Salminen Toni, Salokangas Jari ja Satokangas Lauri, "Paatti-projekti, Projektiraportti", Jyväskylän yliopisto, tietotekniikan laitos, 2012.
- [5] Keränen Tapio, Salminen Toni, Salokangas Jari ja Satokangas Lauri, "Paatti-projekti, Projektisuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, 2012.
- [6] Frankel Nicolas, "Learning Vaadin", Packt Publishing, 2011
- [7] Vaadin Ltd, "Book of Vaadin - 4th Edition", 2012