

Opinnäytetyö (AMK)

Tietojenkäsittelyn koulutus

2020

Samuli Lehtonen

TIETOKANNAN SUUNNITTELU KUNTOUTUSKURSSIEN HAKIJAREKISTERILLE

– Tietokannan suunnitteluprosessi ja sen vaiheet

Samuli Lehtonen

TIETOKANNAN SUUNNITTELU KUNTOUTUSKURSSIEN HAKIJAREKISTERILLE

– Tietokannan suunnitteluprosessi ja sen vaiheet

Tämän opinnäytetyön tarkoituksena oli dokumentoida kuntoutuskurssien hakijarekisteriprojektin suunnitteluprosessin vaiheita ja siihen liitettävän tietokannan suunnittelua. Lopuksi tarkoitus oli luoda suunnitellun tietokannan rakenne. Tavoitteena oli luoda suunnitellun tietokannan rakenne niin, että se on selkeä, turvallinen ja toimeksiantajan vaatimuksien ja toiveiden mukainen. Tavoitteena oli myös dokumentoida rakenteesta kaavio, joka kuvaa visuaalisesti luotua rakennetta. Opinnäytetyössä keskitytään projektin suunnitteluvaiheeseen, siihen liittyviin dokumentteihin ja työkaluihin, sekä hakijarekisterin tietokannan rakenteen suunnitteluun.

Opinnäytetyön alussa tutustuttiin henkilötietoihin, yleiseen tietosuojasetukseen ja sen tuomiin vaatimuksiin henkilötietojen käsittelyssä. Tämän lisäksi avattiin opinnäytetyössä mainittua projektia, sen vaiheita ja dokumentteja, sekä millä tavoin projektinhallinta toteutettiin.

Käytännön toteutuksena suunniteltiin tietoturvaa ja toimeksiantajan vaatimuksia noudattava tietokanta ja dokumentoitiin suunnittelutyön vaiheita. Vaiheisiin kuului tietokannan tietotarpeiden ja datan keräys, rakenteen ja yhteyksien luominen, sekä normalisointi.

Lopputuloksena syntyi kaavio selkeästä, turvallisesta ja toimeksiantajan vaatimuksien ja toiveiden mukaisesta tietokannan rakenteesta. Lopputulosta on tarkoitus hyödyntää työssä mainitun projektin edetessä sen toteutusvaiheeseen. Toteutusvaiheessa tarkoituksena on luoda kuntoutuskurssien hakijarekisterin käyttöliittymä ja tietokanta, jonka rakenne oli tässä opinnäytetyössä tuloksena. Opinnäytetyön tulosta ja siihen johtaneita vaiheita voidaan käyttää apuna vastaavanlaisen tietokannan suunnittelussa.

ASIASANAT:

yleinen tietosuojasetus, tietokanta, projekti, suunnittelu, henkilötieto

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Business Information Technology

2020 | 53 pages, 4 pages in appendices

Samuli Lehtonen

DATABASE PLANNING FOR REHABILITATION COURSE APPLICANT REGISTER

- Stages of project planning process and the database design

The purpose of this thesis was to document the stages of the planning process of an applicant register for rehabilitation courses and the planning of the database which will be attached to the register. Finally, the intention was to create the structure of the planned database as well. The goal was to plan the structure of the database such that it would be clear, secure and fulfill the client's requirements and expectations. The goal was also to document the structure into a diagram which would visualize the created structure. The focus in this thesis is on the planning phase of the project, on the documents and tools related to it, and the planning of the database structure.

At the beginning of the thesis project, it was explored what is understood by personal data and general data protection regulation, and what kind of requirements the regulation brought for handling the personal data. In addition, the mentioned project in the thesis, its stages, documents, and the organisation of the project management were explained.

As a practical implementation, a database was planned which obeyed information security and the requirements the client brought, and the stages of the planning were documented. The stages included collecting the information and data needed for the database, the creation of the structure and connections of the database, and normalization thereof.

A clear and secure diagram of the database structure meeting the client's requirements was created as the final result. The final result is supposed to be utilized as the project advances to development stage. At that stage the user interface and the database, whose structure was developed in this thesis, are to be developed for the register. The results and the stages which lead to the results, may be used as guidance and help to develop a similar database.

KEYWORDS:

general data protection regulation, database, project, planning, personal data

SISÄLTÖ

1. JOHDANTO	7
2. YLEISEN TIETOSUOJA-ASETUKSEN MÄÄRÄÄMÄT VAATIMUKSET HENKILÖTIETOJEN KÄSITTELYSSÄ	9
2.1 Henkilötiedot	9
2.2 Yleinen tietosuoja-asetus ja Suomen Tietosuojalaki	9
2.3 Rekisterinpitäjän velvollisuudet henkilötietojen käsittelyssä	10
2.4 Rekisteröidyn luonnollisen henkilön oikeudet henkilötietojen käsittelyssä	11
3. PROJEKTIN LÄHTÖKOHDAT	13
3.1 Projektin taustaa	13
3.2 Projekti ja sen rakenne	13
3.3 Projektin aloitus	14
3.4 Projektin dokumentit	15
3.4.1 Määrittelydokumentti	15
3.4.2 Salassapitositoumus	16
3.4.3 Tietosuojaseloste	16
3.4.4 Toiminnallinen ja ei toiminnallinen vaatimusmäärittely	17
3.5 Projektinhallinta	19
3.5.1 Scrum	19
3.5.2 Microsoft Teams ja Planner	21
4. TIETOKANNAN SUUNNITTELU	22
4.1 Suunnittelun tärkeys	22
4.2 Tietoturva suunnittelussa	23
4.3 Tietokannan suunnittelun toteutus	24
4.4 Vaatimusten sekä datan kartoitus ja analysointi	24
4.5 Relaatiomalli ja relaatiotietokanta	25
4.6 Tietokannan tietorakenteiden muotoilu	26
4.7 Taululista	27
4.8 Käsittemallit	28
4.9 Käsittemallin toteutus	32
4.10 Taulujen suhteet	39
4.11 Normalisointi	42

5. YHTEENVETO	50
----------------------	-----------

LÄHTEET	52
----------------	-----------

LIITTEET

Liite 1 (1) Projektin määrittelydokumentti

Liite 1 (2) Projektin määrittelydokumentti

Liite 2 (1) Lopullinen ER-Kaavio

Liite 2 (2) Lopullinen ER-Kaavio

KUVAT

Kuva 1. Hakijarekisterin järjestelmän ja tietokannan vaatimukset.	18
Kuva 2. Esimerkki relaatiotietokannan "Staff"-taulusta, sen sisältämistä sarakkeista ja datasta.	26
Kuva 3. Lopullinen taululista.	28
Kuva 4. Esimerkki entiteeteistä ja niiden attribuuteista.	29
Kuva 5. Taulujen välisten suhteiden merkintäviivat.	30
Kuva 6. Esimerkki taulujen suhteiden merkitsemisestä entiteettien välille.	31
Kuva 7. RS-mallin mukaan luotu "Staff"-taulu.	32
Kuva 8. RS-mallin mukaisesti luodut "Hakijat"- ja "Läheiset"-taulut ja niiden sarakkeet.	33
Kuva 9. RS-mallin mukaisesti luotu "Hakemukset"-taulu ja sen sarakkeet.	35
Kuva 10. RS-mallin mukaisesti luotu "Läheisten hakemukset" -taulu ja sen sarakkeet.	36
Kuva 11. RS-mallin mukaisesti luotu Kurssit-taulu ja sen sarakkeet.	37
Kuva 12. RS-mallin mukaisesti luotu "Käyttäjät"-taulu ja sen sarakkeet.	37
Kuva 13. RS-mallin mukaisesti luotu "Maksusitoumukset"- ja "Ensitetopäivähakemukset"-taulut ja niiden sarakkeet.	38
Kuva 14. RS-mallin mukaisesti luodut lokitaulut ja niiden sarakkeet.	39
Kuva 15. Yhteydet ja viiteavaimet hakemustaulujen ja hakijoiden välillä.	40
Kuva 16. Monen suhde moneen -yhteyden muodostaminen yhteystaulun avulla "Hakemukset"- ja "Läheiset"-taulun välille.	41

Kuva 17. "Maksusitoumukset"-taulun ja "Ensitetopäivähakemukset"-taulun välinen	41
Kuva 18. Yhden suhde moneen -yhteydet loki-, "Käyttäjät"-, "Hakijat"- ja "Läheiset"- taulujen välillä.	42
Kuva 19. "Hakijat"- ja "Läheiset"-taulut normalisoitu 1NF	43
Kuva 20. 1NF mukaan normalisoitu "Hakemukset"-taulu ja uudet taulut.	45
Kuva 21. "Läheisten hakemukset" -taulu 1NF normalisoinnin jälkeen.	46
Kuva 22. "Hakemuksien läheiset" -taulu "Hakemukset"-taulun 1NF normalisoinnin jälkeen.	46
Kuva 23. "Kurssit"-taulun normalisointi 1NF muotoon.	47
Kuva 24. "Käyttäjät"-taulu ja sen sarakkeet.	48

1. JOHDANTO

Opinnäytetyön aiheena on tietokannan suunnittelu ja siihen sisältyvät vaiheet. Tietokanta on tarkoitettu suunnitella kuntoutuskurssien hakijarekisteriprojektille, jonka toimeksiantajana on Parkinsonliitto ry. Opinnäytetyössä käydään läpi myös toukokuussa 2018 Euroopan Unionissa (EU) voimaan tullutta yleistä tietoturva-asetusta (GDPR) ja sen tuomia vaikutuksia henkilötietoja käsittelevän tietokannan suunnitteluun ja käyttöön sovel-lusohjelmoijan näkökulmasta.

Ennen tämän projektin alkua Parkinsonliitto ry kirjasi manuaalisesti sähköpostiin tulleet tiedot kurssi-ilmoittautumisista Microsoft Access tietokantasovellukseen. Tämä manuaalinen toimintatapa aiheutti paljon työtä ja prosessia haluttiin nopeuttaa. Lisäksi tiedot siirtyivät selkokielisinä lomakkeelta sähköpostiin luoden tietoturva-aukon. Näistä syistä Parkinsonliitto ry otti yhteyttä theFIRMAan ja antoi toimeksiannon. Toimeksiannon tarkoituksena oli tehdä kurssihakemuksessa annettujen tietojen siirrosta, tarkastelusta ja tallentamisesta tietokantaan automaattisempaa ja helpompaa tietoturvaa unohtamatta.

Päätavoitteena tässä opinnäytetyössä on suunnitella tietokanta hakijarekisterille ja dokumentoida sen rakenne, jonka avulla se voidaan myöhemmin toteuttaa. Tavoitteena on myös suunnitella tietokannasta mahdollisimman selkeä, turvallinen ja toimeksiantajan vaatimuksien ja toiveiden mukainen. Tietokannan tarkoitus on tallentaa hakemuksista ja kurseista syntyvää dataa ja auttaa hallitsemaan kurseille ilmoittautumista.

Tutkimusmenetelmänä opinnäytetyössä on käytetty konstruktiivisia menetelmiä. Ensimmäisessä luvussa kerrotaan mitä henkilötiedot ovat ja miten GDPR:n määräämät asetukset vaikuttavat henkilötietojen käsittelyyn. GDPR:n myötä luonnollisten henkilöiden henkilötietojen käsittelyyn kohdistuva huomio on lisääntynyt ja käsittelystä on tullut paljon tarkempaa, seuratumppaa ja läpinäkyvämpää. Rekisterinpitäjälle kohdistuneista velvollisuuksista ja periaatteiden noudattamisesta on tullut tärkeää ja niiden huomioon ottaminen suunnitteluvaiheessa on tärkeässä roolissa tässä opinnäytetyössä. GDPR:ään tutustuminen tarkkaan ja sen soveltaminen jo projektin suunnitteluvaiheessa on hyvä alku ja tärkeä prosessi onnistuneeseen ja mahdollisimman ongelmattomaan toteutukseen.

Seuraavassa luvussa keskitytään Parkinsonliitto ry:n antamaan projektiin ja sen lähtökohtiin. Luvussa kerrotaan yleisesti mitä projekti tarkoittaa ja minkälaiset vaiheet siinä

on. Projektille valitaan hallintamenetelmä ja luodaan erilaisia dokumentteja, joiden avulla projektia viedään eteenpäin ja tavoitteet saavutetaan.

Viimeisessä luvussa keskitytään itse tietokannan suunnitteluun. Suunnittelu on jaettu opinnäytetyössä kolmeen osaan, jotka käydään läpi vaihe vaiheelta. Nämä vaiheet ovat vaatimusten, sekä datan kartoitus ja analysointi, tietokantarakenteen muotoilu ja tietokannan normalisointi.

Opinnäytetyössä alun perin oli tarkoitus keskittyä suunnittelun lisäksi myös toteutusvaiheeseen. Projektin aikatauluun tuli kuitenkin muutoksia. Projektin työiimin ollessa opiskelijapohjainen heidän löytäminen ja saaminen projektin aloitusvaiheessa viivästytti aloittamista. Lisäksi suunnitteluvaiheen tärkeys lisäsi ajankäyttöä. Suunnitteluun täytyi käyttää paljon aikaa, jotta tietokannan rakenne tulisi olemaan mahdollisimman hyvä ja näin ollen nopeuttamaan myös sen toteutusta. Näiden takia opinnäytetyön suunnittelun aikataulu venyi ja opinnäytetyössä keskityttiin pelkästään projektin tietokannan suunnitteluun ja sen vaiheisiin.

2. YLEISEN TIETOSUOJA-ASETUKSEN MÄÄRÄÄMÄT VAATIMUKSET HENKILÖTIETOJEN KÄSITTELYSSÄ

2.1 Henkilötiedot

Henkilötietoina pidetään kaikkea sitä tietoa, joka voidaan liittää henkilöön suoraan tai välillisesti ja tämän avulla tunnistaa hänet (EU 679/2016, artikla 4 kohta 1). Näitä tietoja on paljon muun muassa luonnollisten henkilöiden etu- ja sukunimet, henkilötunnukset, puhelinnumerot, autojen rekisterinumerot ja potilastiedot. Näitä käytetään helpottamaan luonnollisen henkilön tunnistamista ja vastaavasti myös tunnistautumista.

Mitä yksilöivämpi, tarkempi ja henkilökohtaisempi henkilötieto on, sitä enemmän siitä pitäisi pitää huolta, ettei sitä käytetä väärin. Luonnollisesti sama etu- ja sukunimi voivat löytyä monelta, mutta esimerkiksi tietty henkilötunnus, auton rekisterinumero ja puhelinnumero voi olla vain yhdellä henkilöllä samaan aikaan. Riippumatta käyttökohteesta tai yksilöivämmästä tiedosta, kaikkia henkilötietoja tulee käsitellä tärkeinä, lainmukaisesti, asianmukaisesti ja läpinäkyvästi (EU 679/2016, 5 artikla kohta 1A). Entistä tarkempaa henkilötietojen käsittelystä tulee, kun niitä käsitellään ammatillisessa tai kaupallisessa toiminnassa (Tietosuojavaltuutetun toimisto 2020).

Henkilötietojen turvallisesta käsittelystä ja näin yksityiselämän suojaamisesta sekä yksityisessä että ammatillisessa käytössä on Suomessa määrätty tietosuojalailla, joka kumosi vuonna 1999 käyttöön otetun henkilötietolain. Tietosuojalailla täsmennetään ja täydennetään EU:n asettamaa Yleistä tietosuojaa-asetusta, lyhyesti GDPR ja englanniksi General data protection regulation. (Tietosuojalaki 2018/1050)

2.2 Yleinen tietosuojaa-asetus ja Suomen Tietosuojalaki

Yleinen tietosuojaa-asetus (GDPR) on Euroopan Parlamentin ja Neuvoston asettama asetus 2016/679, jonka tarkoituksena on vahvistaa säännöt luonnollisten henkilöiden suojelulle henkilötietojen käsittelystä, sekä säännöt, jotka koskevat henkilötietojen vapaata liikkuvuutta. Sillä on tarkoitus myös suojata luonnollisten henkilöiden perusoikeuksia ja -vapauksia, sekä oikeutta henkilötietojen suojaan. Sen tavoitteena on parantaa luonnollisten henkilöiden henkilötietojen suojaa ja estää niiden väärinkäyttämistä EU:n alueella. Asetusta sovelletaan osittain tai kokonaan automaattiseen henkilötietojen käsittelyyn, joka tapahtuu EU:n alueella sijaitsevan toimipaikan toiminnan yhteydessä. Se

on tullut voimaan 25. toukokuuta 2018 ja sitä on alettu soveltamaan heti sen voimaantulon jälkeen kaikissa EU:n jäsenvaltioissa. (EU 679/2016, 2016)

Suomen eduskunta hyväksyi asetuksen 5. joulukuuta 2018 ja säätö Tietosuojalain (1050/2018), jonka tarkoituksena on täsmentää ja täydentää luonnollisten henkilöiden suojelusta henkilötietojen käsittelyssä, vapaasta liikkuvuudesta ja GDPR:n soveltamisesta. Tämä laki tuli voimaan 1. tammikuuta 2019 ja se kumosi henkilötietolain (523/1999) sekä myös tietosuojalautakunnasta ja tietosuojavaltuutetusta annetun lain (389/1994). (Tietosuojalaki 2018/1050, 2018)

2.3 Rekisterinpitäjän velvollisuudet henkilötietojen käsittelyssä

GDPR:n myötä henkilötietojen käsittelystä on tullut entistä tarkempaa ja rekisterinpitäjän on myös pystyttävä lain mukaan osoittamaan henkilötietojen käsittelyssä, että siihen koskevat periaatteet täyttyvät (EU 679/2016, artikla 5). Nämä periaatteet ovat seuraavat:

- a. Lainmukaisuus, kohtuullisuus ja läpinäkyvyys, jossa rekisterinpitäjän on käsiteltävä luonnollisen henkilön henkilötietoja lainmukaisesti, asianmukaisesti, sekä läpinäkyvästi henkilötietojen omistajan kannalta.
- b. Käyttötarkoitussidonnaisuus, jossa rekisterinpitäjän on huolehdittava kerätyn henkilötiedon käyttökohteen olevan vain se, mihin sitä on alun perin ollut tarkoitus kerätä. Henkilötietoa ei saa jälkeempään käyttää mihinkään muuhun käyttökohteeseen.
- c. Tietojen minimointi, jossa rekisterinpitäjän on pidettävä käyttökohteeseen kerätyt henkilötiedot vähäisenä ja rajoitettuna määränä ja olla keräämättä tarpeetonta tietoa.
- d. Täsmällisyys, jossa rekisterinpitäjän on huolehdittava kerättyjen henkilötietojen oikeellisuudesta ja täsmällisyydestä. Tietojen on myös oltava päivitetty niiden muuttuessa tai poistettava viipymättä niiden tullessa tarpeettomiksi tai virheelliseksi.
- e. Säilytyksen rajoittaminen, jossa rekisterinpitäjän on säilytettävä henkilötiedot niin, että rekisteröity voidaan tunnistaa niistä vain niiden käsittelyyn tarvittavan ajan.
- f. Eheys ja luottamuksellisuus, jossa rekisterinpitäjän on huolehdittava henkilötietojen käsittelyn asianmukaisesta turvallisuudesta niiden käsittelyn aikana, sekä suojata niitä luvattomalta ja lainvastaiselta käsittelyltä.

- g. Osoitusvelvollisuus, jossa rekisterinpitäjän on pystyttävä osoittamaan edellä mainittujen toimintaperiaatteiden täyttymistä ja toteutumista.

Näiden edellä mainittujen periaatteiden lisäksi rekisterinpitäjän on myös huolehdittava muun muassa siitä, että rekisteröity on antanut selkeästi suostumuksensa henkilötietojen käsittelylle ilmoitettuun toimenpiteeseen (EU 679/2016, artikla 6 kohta 1 A). Tämä suostumus on myös pystyttävä todistamaan. Näiden periaatteiden toteuttamiseksi hyvänä vaihtoehtona on luoda tietosuojaseloste tai sen kaltainen lomake, josta lukija saa vastaukset henkilötietojensa käsittelyyn koskeviin kysymyksiin. Selosteen lukemisen jälkeen hänen on hyvä saada mahdollisuus todistaa selosteen lukeminen ja sen hyväksyminen.

2.4 Rekisteröidyn luonnollisen henkilön oikeudet henkilötietojen käsittelyssä

GDPR:ssä on myös huolehdittu henkilötietojen luovuttajan, rekisteröidyn oikeuksista. Näitä oikeuksia on paljon ja tästä syystä jokainen henkilötietojen luovuttaja ei välttämättä ole niistä tietoinen. Tässä projektissa henkilötietojen luovuttajalla tai rekisteröidyllä tarkoitetaan Parkinsonliiton asiakasta, joka on ilmoittautunut haluamallensa kurssille. Kuten luvussa 2.3 mainittiin, henkilötietojen luovuttajan on annettava suostumus henkilötietojen käytölle. Tästä syystä kurssien ilmoittautumishetkellä ilmoittautujalta on pyydetty suostumus ennen henkilötietojen saamista, jotta henkilötietojen keräilijälle jää todiste suostumuksesta. Kuitenkaan pelkästään suostumus ei täysin riitä antamaan henkilötietojen kerääjälle ja rekisterinpitäjälle oikeuksia tekemään saaduilla tiedoilla mitä tahansa.

Rekisteröidyllä on lain mukaan oikeus saada tietää henkilötietojen luovutushetkenä rekisterinpitäjän identiteetin ja yhteystiedot eli kenelle hän on luovuttamassa henkilötietojaan ja luovuttamien henkilötietojen käyttökohde eli mihin hänen tietojaan tullaan käyttämään, sekä käytön tarkoitus. (EU 679/2016, artikla 13, kohta 1) Näiden lisäksi rekisteröidyllä on oikeus tietää kuka hänen henkilötietojaan käsittelee ja kenelle ne mahdollisesti luovutetaan eteenpäin. Hänellä on myös oikeus saada tietää, jos henkilötietoja tullaan käsittelemään rekisterinpitäjän oikeutetun edun tavoittelemiseksi. (EU 679/2016, artikla 13.)

Tietojen keräyksen jälkeen niitä tullaan myös varmasti säilyttämään toimenpiteessä tarvittavan ajan. Käyttökohteen lisäksi rekisteröity käyttäjä on oikeutettu saada tietää kuinka kauan hänen luovuttamiaan tietojaan tullaan säilyttämään ja käyttämään. Jos tarkkaa aikaa ei pystytä antamaan, täytyy ilmoittaa ajan määrittämiskriteerit, jonka perusteella

tietoja säilytetään tietyn ajan. Rekisteröidyllä on myös oikeus päästä käsiksi henkilötietoihinsa ja saada mahdollisuus pyytää niiden oikaisua. (EU 679/2016, artikla 13 kohta 2a)

Jos rekisteröity ei halua hänen tietojan pidettävän enää rekisterissä hänellä on oikeus pyytää rekisterinpitäjää poistamaan tietonsa, vaikka rekisterinpitäjällä olisi vielä oikeus pitää tietoja rekisterissä. Tästä johtuen on tärkeää suunnitella henkilötietorekisteri ja sen rakenne hyvin, jotta tietoja voidaan myös poistaa helposti ja turvallisesti. (EU 679/2016, artikla 17 kohta 1)

3. PROJEKTIN LÄHTÖKOHDAT

3.1 Projektin taustaa

Projektin toimeksiantajana toimii Parkinsonliitto ry (myöhemmin Parkinsonliitto), joka on kansanterveys- ja vammaisjärjestö. Se on toiminnallaan edistänyt Parkinsonin tautia ja muita liikehäiriösairauksia sairastavien ja heidän omaistensa ja läheistensä hyvinvointia. (Parkinsonliitto, 2020)

Projektin toimeksisaajana on Turun ammattikorkeakoulussa toimiva ICT-projektitoimisto, joka tarjoaa IT-alan palveluja. Toimistossa työskentelee Turun ammattikorkeakoulun henkilöstön lisäksi myös opiskelijoita, jotka ovat myös pääosin projektien työntekijöitä. (theFirma, 2020)

Projektin tavoitteena on suunnitella ja toteuttaa Parkinsonliiton kuntoutuskursseille uusi ilmoittautumislomake, hakijarekisteri ja rekisterin käyttöliittymä. Toteutetun hakijarekisterin tarkoitus on korvata nykyinen hakijarekisteri, joka on rakennettu Microsoft Accessiin ja tiedot siirretään sieltä uuteen hakijarekisteriin. Kuntoutuskursseille ilmoittautuja täyttää Parkinsonliiton nettisivuilla henkilötietonsa lomakkeeseen, jossa on esitätettyinä tarvittavat tiedot kurssista. Nämä täytetyt tiedot menevät suoraan hakijarekisteriin, joka suunnitellaan ja toteutetaan GDPR:ää noudattaen.

3.2 Projekti ja sen rakenne

Projekteja on monenlaisia ja niitä määritellään kirjallisuudessa eri tavoin (Mäntyneva 2016). Ne eivät ole vain mitä tahansa tehtäviä, vaan niille on usein määritetty selkeä tavoite, rajattu resurssit, selkeä alku ja loppu ja se on ajastettu tiettyyn aikaväliin. Tavoitteen saavuttamista ilman projektiin liittyvää byrokratiaa, kuten projektin vaatimusten määrittämistä ja dokumenttien laatimista, ennuste tehtävien onnistumisesta on huono. (Juvonen 2018.)

Projekteilla on rakenne, jota voidaan pitää lähes aina samana kaavana ja pitää sisältään samat vaiheet. Nämä vaiheet eroavat kuitenkin toisistaan toiminnoiltaan, ominaisuuksiltaan ja niissä eri tavoin työskentelyltään. Tämä elinkaari voidaan pilkkoa moneen eri

osaan. Päävaiheina voidaan pitää projektin valmistelu- tai aloitusvaihetta, tämän jälkeistä suunnitteluvaihetta, suunnitelman toteuttamisvaihetta ja lopuksi projektin päätös-vaihetta. (Mäntyneva 2016.)

Tämä projekti on jaettu myös neljään eri vaiheeseen. Ensimmäisessä vaiheessa toimeksiantajalla on syntynyt tarve tuotteelle ja on siitä syystä ottanut yhteyttä toimeksisaajaan, jonka seurauksena projekti on syntynyt ja sen sopimus on solmittu. Sopimuksen allekirjoittamisen jälkeen alkoi toinen vaihe, jossa suunniteltiin hakijarekisterin ja ilmoittautumislomakkeen rakennetta ja kartoitettiin sopivia työkaluja työn toteuttamiseksi. Projektin alun, suunnittelun ja kartoituksen jälkeen alkaa hakijarekisterin rakentaminen ja ilmoittautumislomakkeen implementointi Parkinsonliiton kotisivulle. Viimeisenä vaiheena on hakijarekisterin testaus, mahdollisten virheiden ja bugien korjaus ja tietojen tuonti Microsoft Access:istä. Tässä opinnäytetyössä keskitytään enimmäkseen projektin ensimmäiseen ja toiseen vaiheeseen eli projektin aloitukseen ja tietokannan suunnittelun vaiheisiin. Tästä syystä siinä ei tulla käsittelemään tietokannan ja hakijarekisterin käyttöliittymän toteutuksen ja testauksen vaiheita.

3.3 Projektin aloitus

Projektin alussa toimeksiantaja, Parkinsonliitto, on solminut toimeksisaajan, theFIRMA, kanssa sopimuksen eli toimeksiannon. Siinä theFIRMA saa tehtäväkseen suorittaa jonkin tehtävän toimeksiantajan puolesta. Näiden tehtävien suorittamisprosessia kutsutaan projektiksi. Tässä projektissa projektin osapuolet ovat sopineet kuntoutuskurssien hakijarekisterin toteutuksesta, sen aikataulusta, tavoitteista ja menetelmistä.

Sopimusten ja määrittelyn jälkeen kerätään ja kartoitetaan projektin työryhmä. Tässä projektissa koottu työryhmä tai projektitiimi koostuu projektipäälliköstä, neljästä IT-alan opiskelijasta ja kahdesta ohjaavasta Turun ammattikorkeakoulun opettajasta. Projektitiimin kesken pidettiin aloituspalaveri tai toisella nimellä Kick off -tapaaminen. Tapaamisessa esittäydettiin ja puitiin sitä, mistä projektista oli kyse. Päättämiseksi oli selvittää ensimmäiset toimenpiteet ja tehtävät, joista on hyvä lähteä eteenpäin.

Kick off -tapaamisen jälkeen seuraavana vaiheena oli ottaa yhteyttä Parkinsonliittoon ja sopia heidän kanssaan ensimmäinen tapaaminen. Projektin ensimmäinen asiakastapaaminen on tärkeä. Tapaamisessa usein avataan sopimuksen ja määrittelydokumentin sisältö, sillä pelkästään niiden perusteella on vaikeaa saada kuvaa täsmällisestä tavoit-

teesta. Ajatus osoittautui oikeaksi, sillä tapaamisen aikana tavoite laajentui alkuperäisestä suunnitelmasta, joka on rajattu projektin suunnitelmassa eli määrittelydokumentissa.

Asiakas halusi hakijarekisterin lisäksi käyttöliittymän, josta pystyisi hakemaan rekisteriin kirjattuja ilmoittautuneita, tulostamaan ja selaamaan erilaisia raportteja, sekä lokitietoja. Määrittelydokumentissa ei kuitenkaan ole määritelty tämän kaltaisia toimintoja. Nämä toiminnot osoittautuivat kuitenkin melko tarpeellisiksi ja melkein välttämättömiksi. Dokumentissa mainitaan hakijarekisterin toteutuksesta Parkinsonliiton tarjoamille kuntoutuskursseille, jonka tarkoituksena on yhdistää automaattisesti ilmoittautujat sopiville kursseille. Hakijarekisterin lisäksi projektin sisältöön kuuluu myös ilmoittautumislomakkeen toteutus Parkinsonliiton verkkosivuille. Lopuksi tarkoitus on vielä siirtää vanhasta tietojärjestelmästä tarvittavat tiedot uuteen hakijarekisteriin.

Tapaamisessa painotettiin paljon tietoturvaa ja turvallisen rekisterin luomista. Henkilötietojen käsittelyyn tarkoitettu sovellus tuokin paljon pakollisia vaatimuksia, jotka täytyy täyttää lain mukaan. Suunnittelun alussa onkin tärkeää ottaa huomioon jokainen suunniteltu toiminto ja prosessi tietoturvan kannalta, jotta hakijarekisteri tulee täyttämään vaaditut vaatimukset turvallisella tavalla.

3.4 Projektin dokumentit

Projekteihin liittyy monia tärkeitä dokumentteja, joiden avulla projektin sisältöä, rakennetta ja juridisia tietoja määritetään. Näillä dokumenteilla asetetaan projektille myös tavoite ja määritetään vaatimukset, sekä dokumentoidaan aloitus- ja lopetusvaiheet. Tässä opinnäytetyön projektissa käytetään tavoitteen saavuttamiseksi suunnittelussa ja toteutuksessa hyödyksi määrittelydokumenttia, toiminnallisista ja ei toiminnallisista vaatimuksista koostuvaa vaatimusmäärittelyä, projektisuunnitelmaa, salassapitositoumusta ja tietosuojaselostetta.

3.4.1 Määrittelydokumentti

Tämän projektin sopimuksena toimii Parkinsonliiton ja theFIRMAN luoma projektin määrittelydokumentti, joka löytyy opinnäytetyön liitteenä (Liite 1). Siihen on dokumentoitu tälle projektille tärkeitä tietoja ja ominaisuuksia. Siihen on kirjattu projektin osapuolien

yhteystiedot, sekä yhteyshenkilöt ja projektiin osallistuvat työntekijät, jos he ovat tiedossa sopimuksen luontihetkellä. Dokumenttiin on myös kirjattu projektin alustava aikataulu ja vaiheet, missä järjestyksessä projekti etenee. Näiden vaiheiden aikana on tarkoitus saada dokumenttiin kirjatut projektin tavoitteet täytetyksi. Toteutumisen avuksi dokumentissa on määritelty projektin menetelmät, joissa kerrotaan millä tavoilla projektin vaiheet tullaan toteuttamaan ja avataan vaiheita tarkemmin. Aivan lopuksi dokumentissa rajataan projektin sisältö ja projektiin osallistuvien osapuolien roolit ja tehtävät.

3.4.2 Salassapitositoumus

Tulevaan henkilötietorekisteriin tullaan tallentamaan henkilö- ja terveystietoja. Näitä tietoja projektityöntekijät saattavat nähdä ja saada käsiinsä projektin aikana työskennellessään. Tästä syystä projektiin osallistuvat työntekijän allekirjoittavat salassapitositoumuksen. Salassapitositoumus on tärkeä tehdä ja allekirjoittaa henkilöiden kesken, jotka mahdollisesti joutuvat tekemisiin yksityisen ja salattavan tiedon kanssa.

Tähän projektiin luodussa sitoumuksessa allekirjoittanut sitoutuu olemaan paljastamatta ulkopuolisille mitään, mikä ei ole tarkoitettu julkiseksi ja on laissa määrätty olemaan yksityistä tietoa. Allekirjoittanut sitoutuu myös olemaan väärinkäyttämättä tietoja, joita hän mahdollisesti saa tietoon tai haltuun. Sitoumuksessa luvataan täyttämään vaitiolo- ja salassapitovelvollisuuden myös projektin päätyttyä.

3.4.3 Tietosuojaseloste

Edellisen pääotsikon alla listattiin rekisterinpitäjän velvollisuuksia, tietosuojan periaatteita ja kerrottiin tietosuojaselosteen luomisesta. Selosteen tarkoituksena on tuoda esille henkilötietojen käsittelytoimia ja antaa niille läpinäkyvyyttä. Se on lain mukaan oltava rekisterinpitäjän tai rekisterinpitäjän edustajan ylläpitämä seloste heidän vastuullaan olevista käsittelytoimista. (EU 679/2016, 30 artikla kohta 1.) Sen täytyy olla kirjallinen ja mahdollista löytää myös sähköisenä versiona. Selosteessa tulee olla ilmoitettu rekisterinpitäjän, hänen mahdollisen edustajan ja tietosuojavastaavan nimi ja yhteystiedot, kerättyjen tietojen tarkoitus ja kuvaus rekisteröityjen ryhmistä ja henkilötietoryhmistä. Näiden lisäksi täytyy ilmoittaa mahdolliset kohteet ja ryhmät, joille henkilötietoja luovutetaan ja tietoryhmien poistamisen suunnitellut määräajat, jos se on mahdollista.

Yhtenä listattuna periaatteena on eheys ja luottamuksellisuus, jonka tarkoituksena on turvata henkilötietojen käsittelyn asianmukainen turvallisuus. Tämän täyttämiseksi selosteessa on myös hyvä käydä ilmi millä teknisillä tavoin kerättyjä tietoja turvataan, kuten tietojen salauksesta ja pseudonymisoinnista, sekä miten tiedot tullaan palauttamaan mahdollisen vian sattuessa. Projektiryhmän suunnittelijoiden ja ohjelmoijien on otettava huomioon tulevan rekisterinpitäjän velvollisuudet ja periaatteet hakijarekisteriä suunniteltaessa ja toteutuksessa, jotta ne täyttyvät ja kyseiset tavat voidaan listata tietosuojaselosteeseen rekisteröidyn nähtäväksi.

3.4.4 Toiminnallinen ja ei-toiminnallinen vaatimusmäärittely

Sovellusohjelmoinnissa vaatimuksia määritetään yleensä toiminnallisiksi (Functional Requirements, FR) ja ei-toiminnallisiksi (Non-functional requirements, NFR). Toiminnalliset vaatimukset kuvaavat järjestelmän toiminnallisuutta ja ei-toiminnalliset vaatimukset kuvaavat järjestelmän ominaisuuksia, tarpeita ja rajoituksia (Kurtanović & Maalej 2017). Nämä tulevat myös vaikuttamaan tietokannan rakenteen suunnitteluprosessiin. Järjestelmän toiminnalliset ominaisuudet ovat yleensä herkemmin aiheena suunnittelun aikana ja niihin keskitytään enemmän, ja vasta jälkeempään sovelluskehityksen edetessä ei-toiminnalliset otetaan mukaan (Kurtanović & Maalej 2017). Näitä vaatimuksia kirjataan usein dokumenttiin. Siitä käy ilmi vaatimusten lisäksi niiden tyyppi eli kuuluko se toiminnallisiin vai ei-toiminnallisiin vaatimuksiin ja niiden kohde, eli mihin vaiheeseen sovellusta se kohdistuu.

Tässä projektissa vaatimusmäärittelynä toimii Excel-taulukko, johon on listattu toiminnallisuusryhmä, toiminnallisuuden kohde, vaatimuksen tyyppi ja ID, vaatimuksen kuvaus, sen prioriteetti ja tarpeelliset kommentit ja muistiinpanot asiakkaalta tai projektitiimiltä. Toiminnallisuusryhmä kertoo mihin projektissa tuotettavaan kokonaisuuteen vaatimus kuuluu. Toiminnallisuuden kohteessa kerrotaan, että kuuluuko toiminnallisuus hakijarekisteriin vai julkiseen web-sivuun, johon ilmoittautumislomake tulee. Vaatimuksen tyyppi kertoo, että onko vaatimus toiminnallinen vai ei-toiminnallinen vaatimus ja ID yksilöi vaatimukset toisistaan. Kuvassa 1 näkyy tiivistetyssä muodossa vaatimusmäärittelydokumenttiin luodut toiminnalliset ja ei-toiminnalliset vaatimukset tämän projektin järjestelmälle ja tietokannalle, sekä niiden ID.

PAR_001	Järjestelmän käyttäjänä minulla tulee olla pääsy hakijarekisteriin etäyhteyden kautta, koska joudun hyväksymään kurssille osallistujat kotoa käsin.	PAR_009	Järjestelmän käyttäjänä haluan, että pystyn katsomaan hakijan ilmoittautuslomakkeella lähettämiä tietoja, jotta voin tehdä päätöksen hakijan hyväksymisestä tai hylkäämisestä.	PAR_017	Hakijana haluan, että minulla on mahdollisuus saada tietää miten tietojani säilytetään.
PAR_002	Hakijana tarvitsen www-sivuille ilmoittautuslomakkeen, johon voin syöttää ilmoittautumisen kannalta välttämättömät tiedot tekstinä.	PAR_010	Käyttäjänä haluan saada tulostettua lokitiedostoja	PAR_018	Hakijana haluan, että minulla on mahdollisuus saada tietää mihin tietojani käytetään.
PAR_003	Hakijana tarvitsen www-sivuille ilmoittautuslomakkeen, johon voin syöttää ilmoittautumisen kannalta välttämättömät tiedot pdf-muotoisina liitetiedostoina	PAR_011	Käyttäjänä haluan saada tulostettua tiedot kursseihin osallistuneista ja hyväksytyistä henkilöistä	PAR_019	Hakijana haluan, että minulla on mahdollisuus saada tietää kuka tietojani käyttää.
PAR_004	Hakijana haluan, että minun ei tarvitse syöttää ilmoittautuslomakkeelle sen kurssin tietoja, jolle ilmoittaudun	PAR_012	Käyttäjänä haluan, että kurssille ilmoittautuneiden tiedot ovat mahdollista saada poistetuksi	PAR_020	Hakijana haluan, että hakemukseni lähettämäni tiedot voidaan poistaa pyydettyessä.
PAR_005	Hakijana haluan saada ilmoituksen kurssille hyväksymisestä tai hylkäämisestä.	PAR_013	Käyttäjänä haluan päästä katsomaan hakijan aiempia osallistumiskertoja liiton eri kursseille, jotta voin tehdä päätöksen hakijan hyväksymisestä tai hylkäämisestä haetulle kurssille.	PAR_021	Hakijana haluan, että pyydettyessä saan kopion hakijarekisterissä olevista tiedoistani
PAR_006	Hakijan omaisena haluan että minulta kysytään ilmoittautuslomakkeella eri tietoja kuin hakijalta, koska en halua luovuttaa terveystietojani	PAR_014	Hakijana haluan, että hakemukseni otetaan huomioon kurssin seuraavissa toteutuksissa tasan vuoden ajan hakemuksen jättöpäivästä, jotta minun ei tarvitse hakea kurssin jokaiselle toteutukselle erikseen.		
PAR_007	Järjestelmän käyttäjänä minun pitää pystyä hylkäämään tai hyväksymään hakija kurssille	PAR_015	Käyttäjänä haluan pystyä näkemään tilastoja kursseista/hakijoista ja siirtää ne pdf tai excel muotoon		
PAR_008	Käyttäjänä haluan nähdä tietyille kurssitoteutukselle ilmoittautuneiden/ hakemuksen lähettäneiden hakijoiden nimet	PAR_016	Rekisterinpitäjänä haluan, että tietokanta on suljetussa ympäristössä eikä näkyvillä julkisesti.		

Kuva 1. Hakijarekisterin järjestelmän ja tietokannan vaatimukset.

Toiminnalliset vaatimukset

”Järjestelmään kirjautuminen” tai ”Raporttien tulostaminen” ovat mahdollisia toiminnallisia vaatimuksia. Näillä pystytään kuvaamaan tärkeitä ja vähemmän tärkeitä toiminnallisia vaatimuksia, mitä sovelluksen käyttäjä(t) haluavat ja tarvitsevat. Usein nämä vaatimukset kirjataan yksityiskohtaisempaan muotoon käyttäjätarinaksi, kuten tämän projektin Excel-
taulukossa: ”Järjestelmän käyttäjänä haluan pystyä kirjautumaan järjestelmään, jotta järjestelmää voi käyttää turvallisesti”. Näin toiminnallisuutta voidaan kuvastaa selkeämmin ja tuoda vaadittu toiminnallisuus ilmi ja miksi sitä vaaditaan. Vaatimuksia voidaan myös pilkkoa niin pieniin osiin kuin on tarvetta.

Ei-toiminnalliset vaatimukset

Ei-toiminnallisia vaatimuksia tarvitaan, kun halutaan kuvata kehitettävän sovelluksen ha-
luttuja ominaisuuksia (Ameller ym. 2012). Näihin kuuluu sovelluksen havaittavat ominai-
suudet eli suorituskyky, saatavuus ja käyttövarmuus, sekä sisäiset ominaisuudet eli yllä-
pito ja siirrettävyys alustojen välillä (Ameller ym. 2012). Sovelluksen toiminnallisuus on
usein suunnittelun aikana enemmän pinnalla ja niiden toimimiseen keskitytään paljon ja

enemmän piilossa olevat ominaisuudet jäävät käsiteltäväksi myöhemmin. Nämä ominaisuudet ovat kuitenkin tärkeitä, ellei jopa yhtä tärkeitä kuin toiminnalliset ominaisuudet. Jos tulevassa sovelluksessa tulee olemaan paljon toiminnallisuutta, niin ilman huoliteltua ylläpidon suunnittelua sovellusta ei tulla hallitsemaan kuten on tarkoitus.

3.5 Projektinhallinta

Projektinhallinnan tarkoituksena on muun muassa varmistaa projektin laadullisuus, alen-
taa sen tulevia kustannuksia ja parantaa sen kannattavuutta (Mäntyneva 2016). Projek-
tille valitaan hallintamenetelmä, jonka avulla projektia viedään eteenpäin. Näitä mene-
telmiä on monia ja niiden valinta riippuu muun muassa projektin koosta, työryhmän tai-
doista, budjetista ja projektin luonteesta. Tässä projektissa menetelmänä käytettiin ket-
teryttä hyödyntävää Scrum-menetelmää. Scrumin lisäksi viestintään, projektin työteh-
tävien ja tiedostojen jakoon käytettiin Microsoft Teamsia ja siihen saatavaa lisäosaa
Planneria.

3.5.1 Scrum

Scrum on parhaiten tunnettu ketterän projektinhallinnan menetelmä (Juvonen 2018). Se
on käytetyin ketterän menetelmän viitekehys. Projekti, jossa käytetään scrum menetel-
mää, on mukautuvampi projektin etenemisen aikana syntyviin muutoksiin. Kehityksen
aikana syntyviin muutoksiin ja ongelmiin pystytään vastaamaan paremmin kuin tavalli-
seen ja suoraviivaisempaan projektinhallintamenetelmään, kuten vesiputousmalliin ver-
rattuna. Scrumissa toteutus on silloin hallitumpaa ja nopeampaa. (ScrumAlliance 2020.)

Scrumin on kehittänyt Ken Schwaber ja Jeff Sutherland, ja he ovat myös kirjoittaneet
siitä oppaan, jossa he määrittelevät itse Scrum:in seuraavasti: "Scrum (subst.): Viiteke-
hys, jonka avulla ihmiset voivat ratkaista monimutkaisia ongelmia kehittäessään mah-
dollisimman suurta lisäarvoa tarjoavia tuotteita tuottavasti ja luovasti" (Schwaber &
Sutherland 2017). He määrittelevät myös scrumin olevan kevyt, yksinkertainen ymmär-
tää ja vaikea osata (Schwaber & Sutherland 2017).

Scrumissa projekti pilkotaan pienempiin osiin ja niiden toteutus tapahtuu monissa vai-
heissa, joita kutsutaan sprinteiksi. Projektin alussa voidaan määrittää kuinka monta
sprinttiä tullaan tekemään tai vaihtoehtoisesti suorittaa niitä niin paljon, kuin tarve on.

Nämä sprintit ovat tietyn ajanjakson pituisia, yleensä 1-4 viikon mittaisia, projektin vaiheita tai työkokonaisuuksia, joissa scrum-tiimi työskentelee työtehtävien parissa (Juvonen 2018).

Scrum-tiimi koostuu projektin kehittäjästä, scrumia johtavasta scrummasterista ja tuoteomistajasta, joka on vastuussa projektissa syntyvän tuotteen arvon maksimoimiseen. Käytännössä tuoteomistaja on asiakkaan yhteyshenkilö, joka on projektitiimin tukena projektin aikana. (Schwaber & Sutherland 2017)

Scrumin toteutus tapahtuu tiivistetysti kolmessa osassa, joita sovellettiin myös tässä projektissa. Aivan Scrumin alussa suunnitellaan tulevan tuotteen toiminnallisuuksista koostuva lista. Tässä kohtaa tuoteomistaja on vahvasti mukana vaikuttamassa tuotteen toiminnallisuuksiin ja ominaisuuksiin. Listan luomisen jälkeen alkaa toinen vaihe, jossa projektin kehittäjätiimi suunnittelee tulevan sprintin aikaiset työtehtävät. Työtehtävät jaetaan ja priorisoidaan projektitiimin kesken ja annetaan aika-arvio niiden suorittamiseen. (Schwaber & Sutherland 2017)

Määritetyt tehtävät ja vaatimukset pyritään suorittamaan sprintin aikana. Tästä alkaakin toteutuksen kolmas vaihe eli itse työtehtävien toteutus. Sprintin aikana projektitiimi työskentelee tehtävien parissa jatkaen aina seuraavaan edellisen tullessa valmiiksi. Sprintin aikana pidetään tarpeen vaatiessa ja sprintin pituus huomioiden päivittäin tai viikoittain palaveri. Siinä käydään läpi tämänhetkinen tilanne, mitä kukin on tehnyt sitten viime palaverin, mitä aikoo tehdä seuraavaksi ja onko ilmennyt minkäänlaisia ongelmia (Juvonen 2018). Palaverilla pyritään pitämään tiedonjako nopeana ja tehokkaana (Juvonen 2018).

Jokaisen sprintin päätteeksi kehitystiimillä pitäisi olla esittää jotain toimitettavaa tai toimivaa tuotosta (Juvonen 2018). Kaikkia sprinttiin suunniteltuja työtehtäviä ei kuitenkaan aina saada suoritetuksi sprintin aikana. Tästä huolimatta sprinttiä ei pitkitetä pidemmäksi, vaan esitetään projektitiimin aikaansaannokset. Tämän lisäksi projektitiimi arvioi omaa työntekoa ja listaa tehtävät, joita ei saatu tehtyä sprintin aikana. Nämä uudelleen priorisoidaan ja jatketaan seuraavan sprintin työtehtävien suunnittelusta.

3.5.2 Microsoft Teams ja Planner

Tiedonhallinta ja -kulku ovat tärkeitä käsitteitä projektinhallinnassa. Hyvinkin suunniteltu ja hallittu projekti voi mennä pieleen ilman huoliteltua kommunikaatiota projektin sisällä ja myös ulospäin. (Juvonen 2018) Monessa projektissa käytetään sähköpostia, projektitapaamisia sekä fyysisiä kohtaamisia toimipisteillä että elektronisia keskustelusovelluksia tiedonvälitykseen. Tässä projektissa käytettiin yleiseen kommunikaatioon ja tiedostojen välitykseen sähköpostin lisäksi Microsoft Teams -sovellusta. Microsoft Teams on Microsoftin luoma kommunikaatiosovellus. Sen avulla voi lähettää reaaliajassa viestejä tiimin sisällä ja ulkopuolella, pitää verkkotapaamisia internetin välityksellä ja jakaa tiedostoja. Se myös tukee monenlaisia sovelluksia projektinhallintaan liittyen. Yksi näistä on Planner, jota käytettiin projektissa scrumin toteuttamiseen. Plannerin tarkoitus on organisoida projektin työtehtäviä ja hallita niitä helposti. (Microsoft 2020.) Sen avulla projektin aikana syntyneitä työtehtäviä voitiin jakaa projektitiimin kesken, asettaa niihin prioriteetteja, antamaan aikamääreitä ja informoida projektin kulkua.

4. TIETOKANNAN SUUNNITTELU

4.1 Suunnittelun tärkeys

Tämän projektin keskeisenä asiana oli hakijarekisterin tietokannan suunnittelu. Siksi sen hyvään suunnitteluun kannattaa varata ja käyttää riittävästi aikaa. Mitä enemmän aikaa hyvään suunnitteluun käytetään, sitä enemmän se tuo etuja. (Hernandez 2000) Tietokannat suunnitellaan ja toteutetaan aina jotain tiettyä tarkoitusta varten (Hernandez 2000). Tarkoituksena on usein jonkun ongelman ratkaiseminen tai tietojen käsittelyn parantaminen. Oikean tarkoituksen löytäminen ja määrittäminen on tärkeää suunnittelun kannalta. Se nopeuttaa suunnittelua ja varmistaa, että tietokantaa suunnitellaan oikein täyttääkseen sen päämäärän. Hyvin suunniteltu tietokanta tulee myös ehkäisemään mahdollisten tulevien ongelmatilanteiden syntymistä ja auttaa saavuttamaan tietokannalta vaaditut tavoitteet. (Hernandez 2000) Tietoturvan näkökulmasta hyvin suunniteltu tietokanta pienentää tietoturvariskejä. Se muun muassa pienentää tarvittavien tietokantakyselyiden määrää ja näin vähentäen aikaa, jona tietokantaan ollaan yhteydessä.

Tavoitteet

Tavoitteiden seuraaminen tietokannan rakenteen suunnittelussa auttaa ehkäisemään mahdollisia syntyviä ongelmia, kuten jatkuvat tietokantarakenteen uudelleenmuokkaukset tai vääränlaisten tietojen haut. Näitä tavoitteita on muutama. Niiden seuraaminen suunnittelun aikana auttaa varmistamaan, että tietokannasta pystytään hakemaan vaaditut tiedot ja voidaan suorittaa myös erikoishakuja. Tietoja hakiessa voidaan myös varmistaa, että tiedot ovat eheitä ja niillä on tietokantaa käyttävälle taholle merkitystä. Näin voidaan myös varmistaa GDPR:n mukaisen rekisterinpitäjän velvollisuuden, jossa tallennettujen henkilötietojen täytyy olla eheitä ja täsmällisiä. Nämä tiedot on myös oltava selkeästi tallennettu ja loogisesti nimetty tietokannan eri tauluihin. Lopuksi kannattaa pitää tulevaisuus mielessä ja suunnitella tietokannan rakenne helposti muokattavaksi ja laajennettavaksi. (Hernandez 2000)

Edut

Edellä mainittujen tavoitteiden seuraaminen tuottaa etuja, jotka palkitsevat suunnitteluun käytetyn ajan. Tietokannan valmistuttua ja sen käyttöönoton jälkeen, sitä täytyy myös ylläpitää ja muokata. Kun dataa tulee lisää tai tietokannan rakenteeseen on tehtävä muutoksia, näiden toimenpiteiden suorittaminen tulee olemaan helppoa ja ne eivät tule vaikuttamaan vahingollisesti tietokantaan. (Hernandez 2000) Myös tietokannasta tietojen hakeminen tulee olemaan helppoa, sillä taulujen rakenne ja niiden väliset suhteet ovat kunnossa. Lopuksi isona etuna on ajan säästäminen tietokannan ympärille tulevien käyttöliittymien ja sovellusten ohjelmoinnissa.

4.2 Tietoturva suunnittelussa

Henkilötietoja tallentavan tietokannan suunnittelussa tietoturva on tärkeä käsite. Tietokanta suunnitellaan sen tulevalle omistajalle. Tietokanta tuleeikin siksi suunnitella niin, että hänen on pystyttävä täyttämään GDPR:ssä vaaditut rekisterinpitäjän velvollisuudet. Nämä velvollisuudet listattiin aikaisemmin luvussa kaksi.

Käyttötarkoitussidonnaisuus ja tietojen minimointi luovat pohdintaa rakenteen suunnitteluun. Käyttötarkoitussidonnaisuuden mukaan henkilötietojen käyttö kohdistuu ainoastaan siihen, mihin niitä on alun perin tarkoitettu käytettäväksi. Tästä johtuen tietokannan rakenne ja suhteet täytyy suunnitella niin, että ne palvelevat juuri sitä käyttökohdetta, johon se ja sinne tallennettavat henkilötiedot ovat tarkoitettu. Tässä projektissa tietokantaan tallennettavien henkilötietojen käyttökohde on kurssi-ilmoittautumisten ja -hakemusten käsittely. Lisäksi tietoja on tarkoitus käyttää erilaisten raporttien luomiseen muun muassa rahoittajille tai muille toimintaan osallistuville tahoille kuten esimerkiksi Kelalle.

Tietokantaan tallennettavien tietojen minimoinnissa täytyy tarkkaan määrittää tietojen tarpeellisuus. Ylimääräisiä ja tarpeettomia henkilötietoja ei saa tallentaa. Suunnitellessa tietokantaa onkin tärkeää määrittää tarkkaan tietokantaan tallennettavat datat, jotta ne myös palvelevat rekisterin tarkoitusta.

4.3 Tietokannan suunnittelun toteutus

Tietokantojen suunnittelun toteutukseen on monia eri menetelmiä ja metodeja, mutta suunnitteluprosessin aikana suoritettavat vaiheet on hyvä opetella. Tässä projektissa tietokannan suunnittelumenetelmänä käytettiin perinteiseen menetelmään pohjautuvaa prosessia. Tähän menetelmään kuuluu kolme vaihetta: Vaatimusten, sekä datan kartoitus ja analysointi, tietokantarakenteen muotoilu ja tietokannan normalisointi. (Hernandez 2000)

4.4 Vaatimusten sekä datan kartoitus ja analysointi

Ensimmäisen vaiheen tarkoitus on selvittää kaikki tarvittava data, jota tarvitaan tietokannan rakenteen muodostamiseen, sekä analysoidaan nykyisen tietokannan rakennetta. Vaiheen tarkoitus on myös selvittää tietokannan vaatimukset ja tietokantaan tallennettavan datan tarkoitus eli luoda selkeä syy datan tallennukselle ja näin ollen helpottaa tulevaa rakenteen luomisvaihetta. Datan ja tietotarpeiden kartoitus on tärkeää, sillä GDPR:n mukaan mitään henkilötietoja ei saa tallentaa ilman selkeästi osoitettavaa tarkoitusta (EU 679/2016, artikla 5 kohta 1b).

Tämä vaihe alkoi jo projektin alussa, jolloin projektia määriteltiin. Silloin Parkinsonliitolta tiedusteltiin heidän tietotarpeita ja toimeksiannossa selostetun uuden hakijarekisteriin dataa. Alun perin tarkoitus oli käyttää samoja tietoja, mitä sen hetkisessä hakulomakkeessa oli pyydetty hakijalta. Tämä muuttui melko nopeasti, kun Parkinsonliitto alkoi kartoittamaan tarvittavia tietotarpeita. Tarvittavat tiedot pienentyivät jonkin verran prosessin aikana.

Tiedot lopulta koostuivat pitkälti hakijan tavallisista henkilötiedoista, kuten nimestä, henkilötunnuksesta, puhelinnumerosta, sähköpostiosoitteesta ja lähiosoitteesta, mihin kurssiin on ilmoittautunut ja siihen liittyviä tietoja, sekä hakijan terveystiedoista ja niihin liittyvistä lisätiedoista. Terveystietoihin kuului muun muassa diagnosoitu sairaus, miten se vaikuttaa hakijan elämään ja muut mahdolliset sairaudet. Näiden lisäksi tarve oli tallentaa tietoa hakijan läheisestä, joka osallistuisi kurssille hakijan kanssa. Hänestä tarvittiin samat henkilötiedot kuin hakijastakin. Näiden lisäksi läheisiltä tarvittiin tieto hakijan ja läheisen välisestä suhteesta, onko läheisellä omia sairauksia, minkälainen toimintakyky hänellä on ja tarvitseeko hän apua missään toiminnassa. Läheisillä oli myös mahdolli-

suus hakea heille tarkoitetuille kursseille yksin, joten myös heidän lähettämistään kurssihakemuksista täytyi tallentaa tietoa. Näihin tietoihin kuului myös tavalliset henkilötiedot nimestä lähiosoitteen ja lisäksi selostus henkilön, jonka takia läheinen hakee kurssille, sairauksista, diagnoosista, muista sairauksista ja rajoitteista.

Tämän avuksi luotiin dokumenteissa mainittu toiminnallinen ja ei toiminnallinen vaatimusmäärittely auttamaan datan ja tietotarpeiden määrittämisessä. Vaatimukset auttoivat havainnollistamaan tulevan järjestelmän vaatimuksia ja mitä tietoja tietokannasta täytyi saada ja näin ollen se auttoi kertomaan mitä dataa sinne täytyi tallentaa. Projektin ohjelmoijat loivat heidän näkökulmastaan ensimmäisen version vaatimusmäärittelystä. Se lähetettiin Parkinsonliitolle kommentoitavaksi ja muokattavaksi, jonka avulla viimeinen versio tehtiin yhdistämällä ensimmäinen ja Parkinsonliiton versio.

Luodusta vaatimusmäärittelystä pystyttiin poimimaan muutama tietotarve, jotka auttavat myöhemmin tietokantarakenteen luomisessa. Yhtenä vaatimuksena oli ”Hakijana tarvitseen www-sivuille ilmoittautumislomakkeen, johon voin syöttää ilmoittautumisen kannalta välttämättömät tiedot tekstinä.” Siitä saatiin poimittua tietotarpeiksi hakija ja ilmoittautumislomake eli hakemus. Toinen vaatimus koski hakijoiden läheisiä ja heidän henkilötietoja. Tietotarpeisiin voitiin lisätä läheiset.

Parkinsonliitolta saatiin myös nykyisen tietokannan rakenteesta kopio, joka oli luotu käyttäen relaatiomalliin pohjautuvaa relaatiotietokannan hallintajärjestelmää Microsoft Accessia. Kopio esitti tietokannan käyttöliittymää ja havainnollisti sen toimintoja, mutta kuitenkin toteuttamatta kunnollisia ”structured query language” -kielen (SQL) hakuja sen ollessa vain kopio ilman yhteyttä oikeaan tietokantaan. Se sisälsi tietokantataulut ja datasarakkeet ilman mitään henkilötietoja, jotta tietoturva ei rikottaisi. Tämä kopio auttoi havainnollistamaan heidän tarpeitaan ja minkälaisen tietokannan he tarvitsivat. Tietokannan kopiosta saatiin myös apua uuden tietokannan rakenteen suunnitteluun.

4.5 Relaatiomalli ja relaatiotietokanta

Projektin tietokantatyypiksi muodostui SQL-pohjainen relaatiomallia noudattava relaatiotietokanta. Mallin on perustanut E. F. Codd vuonna 1970, ja se korvasi aikaisemmat hierarkkiset ja verkkomalliset tietokantatyypit (Hovi ym. 2005). Relaatiomalli perustuu muun muassa Joukko-oppiin. Siinä tietokannan data muodostaa joukon rivejä, joihin voidaan kohdistaa hakuja käyttäen erilaisia joukko-operaattoreita. Näitä voidaan toteuttaa käyttäen SQL-kieltä.

4.6 Tietokannan tietorakenteiden muotoilu

Vaatimusten ja datan kartoituksen ja analysoinnin jälkeen voidaan aloittaa suunnittelun toinen vaihe eli relaatiotietokannan tietorakenteen muotoilu. Tietokannan rakennetta kuvaillaan relaatiomallissa usein tauluilla. Taulut jakavat tietokantaan tallennettavat datat omiin osioihin, joiden sisältö koostuu sarakkeista tai toisella nimellä kolumneista. Sarakkeet taas täyttyvät datalla, jotka muodostavat tauluun rivejä. (Hovi ym. 2005.)

Kuvassa 2 on yleinen visuaalinen esimerkki relaatiotietokannan taulun rakenteesta. Siinä taulun nimi on ylimpänä ja sarakkeiden nimet suoraan sen alla vaakarivillä alkaen ID:stä loppuen arvonimeen. Sarakerivin alle tulee datarivit, jotka koostuvat yksittäisistä datakentistä ja niihin sijoitetuista arvoista.

Staff			
ID	first_name	last_name	title
1	Taisto	Tietokanta	Terapeutti
2	Liisa	Lintta	Lääkäri
3	David	Daattaaja	Diagnosoija
4	Hermann	Huomaaja	Hoitaja
5	Kaapo	Kärkkäinen	Kirurgi
6
7

Kuva 2. Esimerkki relaatiotietokannan "Staff"-taulusta, sen sisältämistä sarakkeista ja datasta.

Tauluille annetaan luomishetkellä yksilöivät nimet. Tietokannan taulujen nimet on usein hyvä kirjoittaa englanniksi. Se helpottaa muun muassa muunkielisiä ohjelmoijia, jotka mahdollisesti liittyvät projektiin myöhemmin ja helpottaa ohjelmoimista, kun voidaan käyttää kahden kielen sekoittamisen sijaan vain englannin kieltä.

Nimeäminen voi olla vaikeampaa mitä aluksi luullaan, koska nimen täytyy olla mahdollisimman tarkka ja kuvaava, mutta kuitenkin ei kannata tehdä ylipitkää nimeä. Nimestä täytyy myös ymmärtää, mitä dataa taulu pitää sisällään, joten myöskään lyhenteitä ei kannata käyttää nimenä. Ne voivat luoda hämmennyksiä taulun tarkoitusta miettiessä ja eivät kerro tarkasti mitä tauluun on tallennettu. Taulun nimet laitetaan monikkomuotoon, koska taulu koostuu siihen liittyvän datan kokoelmasta ja nimen täytyy viitata kaikkiin niihin tietoihin. (Hernandez 2000)

Kuvan 1 taulun nimeksi on annettu ”Staff” eli henkilökunta. Sen nimi on selkeä, eikä liian pitkä ja jo pelkän nimen perusteella voi tietää kyseisen taulun pitävän tietoa esimerkiksi yrityksen tai jonkin tapahtuman henkilökunnasta. Taulu on kopio tämän projektin suunnitellusta tietokannasta ja tässä tapauksessa siihen tallennetaan kurssisiin osallistuvan henkilökunnan tietoja.

Jokaisen taulun sisällä on siihen liittyvää dataa, jotka ovat jaettuna omiin sarakkeisiin. Nämä sarakkeet määritetään ja niille annetaan nimi ja tyyppi, jotka kuvastavat yksittäistä tallennettua dataelementtiä taulussa. Sarakkeet nimetään yksikössä, koska ne kuvastavat tiettyä dataelementtiä. (Hernandez 2000)

Taulun sarakkeisiin tallennetut dataelementit muodostavat keskenään rivejä, joissa jokaisen sarakkeen kohdalle tallennetaan yksi dataelementti. Nämä rivit toistuvat ja ne yksilöidään omalla tunnisteella, joka on usein nouseva luku tai muu yksilöivä eli uniikki tieto. (Hovi 2013.)

4.7 Taululista

Tietokannan suunnittelun ensimmäisessä osassa kartoitettiin Parkinsonliiton tietotarpeita ja vaadittua dataa, mitä tietokantaan täytyi tallentaa. Näistä tiedoista luotiin tiedosto, jota voidaan kutsua taululistaksi (Hernandez 2000). Taululistan avulla pystytään hahmottamaan ja luomaan alustavasti tietokannan taulut ja niiden sarakkeet (Hernandez 2000). Alustavalla tarkoitetaan, että taululistaan luodut taulut tulevat todennäköisesti muuttumaan suunnittelun aikana, sekä tauluja luodaan mahdollisesti lisää. Listan tarkoitus on olla pohjana rakenteen aloitukseen. Sitä pystyy myös nopeasti muokkaamaan tarvittaessa, ennen kuin aloitetaan luomaan tietokannasta itse kaaviota ja normalisoimaan.

Taululistaan listattiin tietotarpeista alustavien taulujen nimet. Ensimmäisten taulujen nimiksi muodostui ”Hakijat” ja ”Hakemukset”. Näiden lisäksi tarve oli tallentaa myös hakijoiden läheisten tietoja, joten listaan lisättiin ”Läheiset”-taulu. Hakijoiden läheisille oli tarjolla heille kohdistettuja kurssveja, joten läheisten omille hakemuksille täytyi luoda oma

taulu, "Läheisten hakemukset". Hakijoiden ja heidän läheistensä hakemukset kohdistuvat johonkin kurssiin, joten kurssit tarvitsivat oman "Kurssit"-taulun. Viimeisenä lisäyksenä taululistaan oli "Käyttäjät"-taulu Parkinsonliiton työntekijöille. Tämän taulun tarkoitus on pitää kirjautumistietoa järjestelmän käyttäjistä. Lopulliseksi taululistaksi muodostui kuvan 3 mukainen lista. Tätä luotua listaa tullaan käyttämään apuna käsitemallien luomisessa.

APPLICANTS
APPLICATIONS
RELATIVES
RELATIVE APPLICATIONS
COURSES
USERS

Kuva 3. Lopullinen taululista.

4.8 Käsitemallit

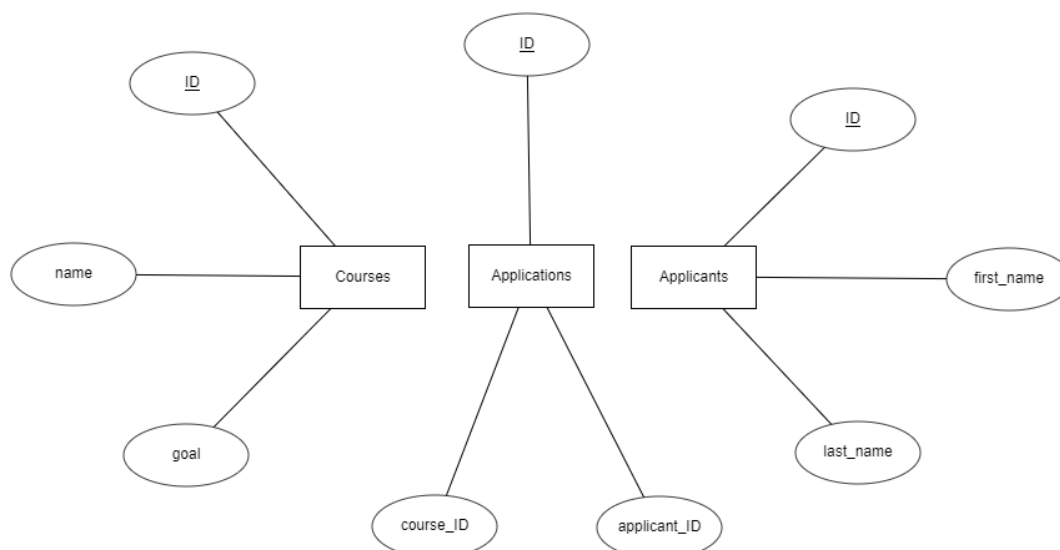
Taulujen ja niihin tallennettavan datan hahmottamiseen ja taulujen välisten suhteiden määrittämiseen visuaalisesti voidaan käyttää apuna erilaisia käsitemalleja ja -kaavioita. Tässä projektissa tietokannan rakennetta suunnitellessa käytettiin käsittekaaviota, Entity Relationship -diagrammia eli ER-kaaviota (ERD) ja siihen liittyvää relaationaalinen skeema -kaaviota (Relational schema, RS). Datana kaavion luomiseen käytettiin vaatimusten, sekä datan kartoitus ja analysointi vaiheessa kerättyä dataa ja edellisessä osiossa luotua listaa tauluista.

Entity Relationship -Diagrammi

Yhtenä menetelmänä on hyvin tunnettu ER-käsitemalli (Entity relationship). Tämän menetelmän esitti Peter Chen vuonna 1976 (Hovi & Huotari & Lahdenmäki 2005). Menetelmässä on tarkoitus kuvata kohdealuetta eli tietokannan rakennetta visuaalisesti käyttäen käsitteitä, niiden ominaisuuksia ja käsitteiden välisiä suhteita. Se on tehokas tapa, jolla

voi mallintaa visuaalisesti tietokannan rakennetta. Siinä muodostetaan visuaalisesti entiteetit (Entity) eli taulut ja niihin liitetään attribuutit, jotka kuvastavat tauluihin tallennettavaa dataa.

Esimerkkinä ER-käsitelmästä voidaan käyttää tästä projektista otettua kolme entiteettiä ja yhdeksän attribuuttia. Entiteetit "Courses"(Kurssit), "Applications"(Hakemukset) ja "Applicants"(Hakijat) kuvaavat tietokannan tauluja. Näihin jokaiseen entiteettiin on liitetty kolme attribuuttia, jotka kuvaavat sarakkeita. Kappaleessa 4.6 mainittiin, että jokaiselle entiteetille annetaan yksilöivä tunniste eli pääavain, joka on tässä tapauksessa alleviivattu ID-attribuutti. Normaaleja attribuutteja Kurssit-entiteetillä on "name"(nimi) ja "goal"(tavoite) ja Hakemukset-entiteetillä "first_name"(etunimi) ja "last_name"(sukunimi). Kuvassa 4 näkyy tämän esimerkin entiteetit ja attribuutit.



Kuva 4. Esimerkki entiteeteistä ja niiden attribuuteista.

Hakemukset-entiteetille on liitetty sen ID:n lisäksi Kurssit- ja Hakijat-entiteettien ID-attribuutit. Näitä kutsutaan viiteavaimiksi ja ne yhdistävät taulut keskenään luoden niiden välille loogisen yhteyden eli suhteen (Relationship). Suhdetyyppejä kuvataan ERD:ssä usein viivoilla, jotka liittyvät taulusta tauluun. Näitä tyyppisiä on yhden suhde yhteen, yhden suhde moneen ja monen suhde moneen.

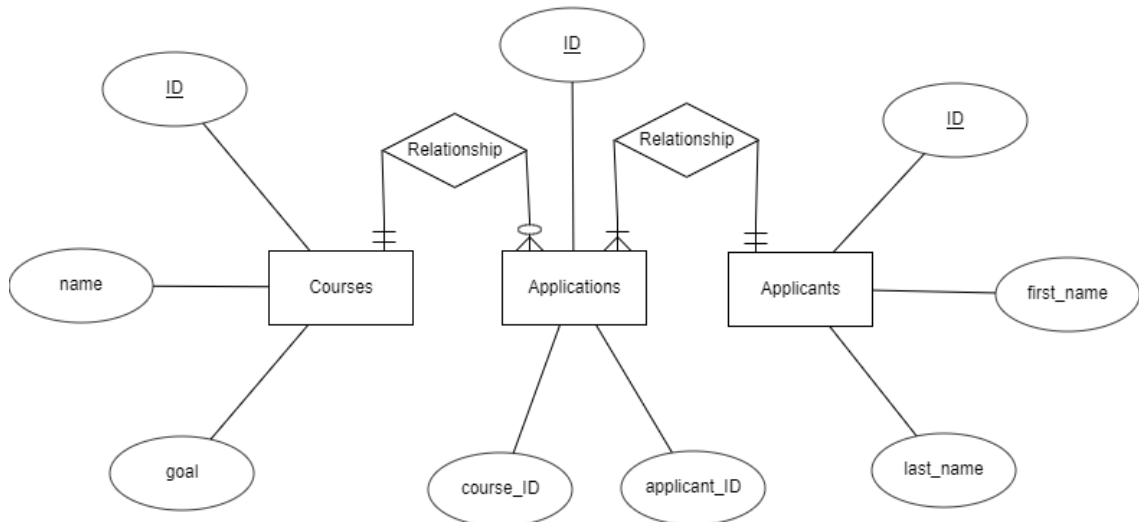
Yhden suhde yhteen on melko yksinkertainen suhde. Siinä ensimmäisen taulun yksi rivi liittyy ainoastaan yhteen toisen taulun riviin ja sama toisinpäin. Yhden suhde moneen -yhteydessä ensimmäisen taulun perspektiivistä yksi rivi liittyy ainoastaan yhteen toisen taulun riviin, mutta toisen taulun perspektiivistä yksi rivi voi liittyä yhteen tai useampaan riviin ensimmäisessä taulussa. Näiden oikeinpäin laittamisessa pitää olla tarkkana, jotta

ne kuvaavat entiteettien välistä suhdetta oikein. Monen suhde moneen -yhteydessä molempien taulujen rivit voivat liittyä moneen toisen taulun riviin. Näiden suhteiden viivat ovat esitetty kuvassa 5 Ylimpänä kuvassa on yhden suhde yhteen -yhteyttä kuvaava viiva, jonka molempiin päihin laitetaan poikittainen viiva. Toinen viiva kuvastaa yhden suhde moneen -yhteyttä. Tähän viivaan toiseen päähän laitetaan poikkiviiva, ja toiseen kolmiosainen haaraviiva. Viimeisenä suhteena on monen suhde moneen, jonka molempiin päihin laitetaan kolmihaarainen viiva.



Kuva 5. Taulujen välisten suhteiden merkintäviivat.

Kuvassa 6 on "Kurssit"- ja "Hakemukset"-entiteetit liitetty toisiinsa ja suhde on yhden suhde moneen. Kurssit-entiteetin näkökulmasta yksi kurssirivi voi liittyä moneen hakemusriviin. Kolmihaarain yläpuolella on soikio, joka mahdollistaa, että Kurssit-entiteetin rivi ei välttämättä liity yhteenkään hakemukseen. Tämä on mahdollista tilanteessa, kun kurssi luodaan ja ainuttakaan hakemusta ei ole tehty luodulle kurssille. Hakemus-entiteetin näkökulmasta yksi hakemus voi kohdistua yhteen ja vain yhteen kurssiin, koska yhdellä hakemuksella voidaan hakea vain yhteen kurssiin. Tämä "yhteen ja vain yhteen" väite voidaan merkitä kahdella poikkittaisella viivalla. Hakijat- ja Hakemukset-entiteetit ovat myös liitetty toisiinsa suhteella yhden suhde moneen. Yksi hakemus voi liittyä vain yhteen hakijaan ja hakija liittyy yhteen tai moneen hakemukseen, koska hakijasta ei olisi muuten tietoa, ellei hän ole hakenut kurssille. Hakija pystyy myös tekemään monta hakemusta ja siksi myös liittymään moneen hakemukseen.



Kuva 6. Esimerkki taulujen suhteiden merkitsemisestä entiteettien välille.

Vaihtoehtoisesti toinen tapa mallintaa tietokannan rakennetta ERD:in lisäksi on tässä projektissa käytetty ”relaationaalinen skeema” -mallinnus tapaa (englanniksi relational schema, RS). Erona ERD:iin, jossa keskitytään enemmän taulujen välisiin suhteisiin ja datan visualisointiin, RS:ssa keskitytään enemmän datan tallennuksen rakenteeseen ja minkä tyyppistä dataa tauluihin tallennetaan. RS:ssa taulut muodostetaan käyttämällä taulukkomallia.

Kuvassa 7 on esimerkki RS-mallin taulusta nimeltä Staff, joka on kuvan 2 tietokantataulun RS-mallin mukainen rakenne. Taulun nimen alla vasemmalla on listattuna sarakkeiden nimet ja oikealla tietotyypit. Nämä sarakkeet määrittävät tauluun tallennettavan datan nimen ja kertovat minkä tyyppinen data on. Tässä taulussa yksilöivänä sarakkeena toimii ID, joka toimii myös taulun pääavaimena ja on merkitty PK (Primary Key)-lyhenneellä. Sen sisältämä data on aina eri joka rivillä. Se voi myös koostua useammasta sarakkeesta. (Hovi 2013.) Usein pääavaimena toimii kuitenkin yksi sarake ja se tietotyyppinä nouseva kokonaisluku, mutta voi olla muu yksilöivä tunniste kuten käyttäjillä sähköpostiosoite. Tietotyyppi ilmaistaan sarakkeen nimen oikealla puolella ja tässä tapauksessa sanalla Integer eli kokonaisluku. Tämän lisäksi sulkeisiin on määritetty ID:n maksimipituus, jonka mukaan tietokanta varaa muistia dataelementille. ID:n lisäksi tässä kuvan 6 taulussa on listattuna ”etunimi”-, ”sukunimi”- ja ”arvonimi”-sarakkeet. Näihin tallennetaan niiden nimeä vastaavat datat. Tässä tapauksessa niihin tallennetaan kurseihin

osallistuvien henkilökunnan etu- ja sukunimi ja arvonimi. Ne ovat myös määriteltäviä olemaan vaihtelevan pituisia merkkijonoja ja maksimissaan 40 merkin pituisia.

Staff	
ID	Integer(4)(PK)
first_name	Varchar(40)
last_name	Varchar(40)
title	Varchar(40)

Kuva 7. RS-mallin mukaan luotu "Staff"-taulu.

4.9 Käsitellin toteutus

RS-mallin mukaista käsittekaaviota aloitettiin työstämään luodun taululistan pohjasta. Kaavion luomiseen voidaan käyttää internetistä löytyviä ERD- ja RS-diagrammin luomiseen tarkoitettuja sovelluksia. Tässä projektissa käytettiin Lucidchart-sivuston tarjoamaa työkalua kaavion luomiseen. Sivusto tarjosi puitteet sekä ERD että RS-diagrammin luomiseen ja mahdollisti myös samanaikaisen työskentelyn usean henkilön kanssa samanaikaisesti.

Hakijat- ja läheiset-taulut

Ensimmäisenä RS-kaavioon lisättiin "Hakijat"- ja "Läheiset"-taulut. Näiden taulujen tarkoitus on kerätä kurssihakemuksissa lähetetyt perustiedot varsinaisista hakijoista (myöhemmin hakijat) eli sairastavista osapuolista ja heidän läheisistään. "Läheiset"-tauluun tullaan kirjaamaan myös tiedot läheisten omille kursseille hakevista. Taulujen sarakkeiden muodostaminen aloitettiin kirjaamalla ensimmäisenä "ID"-sarake, joka yksilöi taulun rivit. ID toimii pääavaimena ja on automaattisesti nouseva kokonaisluku. Aina uuden rivin muodostuessa, ID nousee yhdellä kokonaisluvulla. Seuraavaksi hakemukseen kirjatut perushenkilötiedot lisättiin tauluihin, joista ensimmäisenä lisättiin hakijan ja läheisen nimet. Nimen lisäksi tauluun lisättiin hakijan ja läheisen henkilötunnus. Henkilötunnus täytyy kirjata muun muassa Kelan vaatimana. Näiden lisäksi hakijoihin ja läheisiin koettiin

liittyvän heidän puhelinnumero, sähköposti ja kotiosoite. Kaikki sarakkeet nimestä osoitteeseen merkittiin vaihtelevan pituisiksi teksteiksi, koska ne sisältävät tekstimuodossa olevaa tietoa ja mahdollisesti myös kokonaislukuja. Edellä mainittujen lisäyksistä syntyi ensimmäinen versio "Hakijat"- ja "Läheiset"-taulusta, jotka näkyvät kuvassa 8.

Applicants		Relatives	
ID	Integer(4)(PK)	ID	Integer(4)(PK)
name	Varchar(255)	name	Varchar(255)
identity_number	VarChar(11)	identity_number	VarChar(11)
phone	VarChar(14)	phone	VarChar(14)
email	Varchar(255)	email	Varchar(255)
address	Varchar(255)	address	Varchar(255)

Kuva 8. RS-mallin mukaisesti luodut "Hakijat"- ja "Läheiset"-taulut ja niiden sarakkeet.

Hakemukset- ja Läheisten hakemukset -taulut

Seuraavaksi siirryttiin hakemustauluihin. Nämä taulut suunniteltiin olevan tietokannan keskiö, johon suurimmat osat hakemuksista tulleet tiedot tallennetaan ja hakemuksiin viittaavat tiedot kohdistuvat. Tauluja täytyi tehdä kaksi, vaikka tauluihin tallennettiin keskenään paljon samantyylistä tietoa. Keskeisimmät erot olivat hakemuksen tekijät ja kuinka paljon tietoa heistä tarvitaan. Läheisten tekemistä hakemuksista tallennettiin vähemmän tietoja, kuin kuntoutuvat hakijan hakemuksista, koska siihen ei sisältynyt kaikkia tietoja sairastavasta osapuolesta, jonka takia läheinen hakee kurssille. Tästä syystä tauluun, johon tallennettaisiin sekä hakijan että läheisen tekemät hakemukset, tulisi paljon tyhjiä dataelementtejä eri sarakkeisiin.

"Hakemukset"-taulun ensimmäinen sarake oli luonnollisesti yksilöivä "ID"-sarake. Tämän lisäksi tauluun tallennettiin hakemuksen aikaleima, jolloin hakemus on jätetty. Tämä tieto auttaa hakemuksien voimassaolon laskemiseen ja lajitteluun. Hakemuksilla oli myös oma tila, jota täytyi myös pystyä asettamaan ja muokkaamaan. Näitä tiloja oli ei hyväksytty, hyväksytty, varalla, peruttu ja voimassa. Näiden elementtien tyyppiksi asetettiin to-tuusarvo, joka on joko tosi tai epätosi. Hakemuksien tilaa täytyi myös pystyä kommentoimaan esimerkiksi miksi hakemus hylättiin tai asetettiin varalle.

Hakemuksessa kysyttiin kuntoutuvan hakijan sairauksiin liittyviä tietoja, jotka lisättiin ”Hakemukset”-tauluun. Sairaudesta kysyttiin hakemuksessa itse diagnosoidun sairauden nimen lisäksi sen tarkennetta ja diagnosoinutta lääkäriä, sekä sairauden ilmenemisvuotta ja -kuukautta. Myös tietoa sairauden vaikutuksesta elämään ja ensioireiden ajankohdasta tallennettiin tauluun. Kaiken näiden lisäksi hakijalta pyydettiin tietoa muista mahdollisista sairauksista, mitkä voisivat vaikuttaa kurssiin. Viimeisenä hakijoiden sairauksiin liittyvää tietoa oli mahdollisen sairaalabakteerin nimi ja saamisvuosi. Tämä tieto oli tarpeen tallentaa vain niiltä hakijoilta, jotka tulivat myöhemmin valituksi haetulle kurssille. Hakijoille haluttiin myös antaa mahdollisuus lisätä hakemukseen lisätietoja, jonka avulla he voivat tarkentaa jo annettuja tietoja.

Hakijoiden toimintakykyyn viittaavia asioita oli myös tarve tallentaa tietokantaan. ”Hakemukset”-tauluun tallennettiin heidän avuntarve erilaisissa elämäntilanteissa ja toimissa, mahdolliset apuvälineet, työllisyyden tilanne ja nykyinen tai entinen työ. Näiden lisäksi hakijoiden allergiat ja ruokavaliot olivat tarpeen tallentaa tietokantaan. Näitä tietoja ei kuitenkaan kysytty hakemuksessa, sillä ne kysyttiin vain niiltä hakijoilta, jotka tulivat myöhemmin valituksi haetulle kurssille.

Hakijoiden oli mahdollisuus lisätä kurssihakemukseen läheisiä, joiden perustiedot menivät aikaisemmin mainittuun ”Läheiset”-tauluun. Läheisistä haluttiin tallentaa perustietojen lisäksi heidän suhteensa hakijaan, työllisyyden tilanne, mahdolliset sairaudet, ruokavaliot ja heidän mahdolliset avuntarpeensa.

Hakijoilta haluttiin myös tiedustella mistä he saivat tietää kurssista, johon he hakivat. Heiltä ja heidän läheisiltään haluttiin myös tallentaa tavoitteet haetulle kurssille. Näiden lisäksi Parkinsonliitto halusi tietää hakijoiden aikaisempien kurssien tietoja. Nämä tiedot koostuivat kurssin nimestä, päivämäärästä ja sijainnista.

Parkinsonliitto halusi myös mahdollisuuden kirjata omia muistiinpanoja hakijasta. Näiden kommenttien tallentaminen päätettiin sijoittaa ”Hakemukset”-tauluun, sillä kommentit kohdistuivat hakijaan, joka on hyväksytty hakemuksessa haetulle kurssille.

Lopuksi hakijoista haluttiin tallentaa ennakkotietoa hakijan tietoteknisistä laitteista ja yhteyksistä, jos hän osallistui verkkokurssille. Näitä tietoja oli hakijan käyttämien laitteiden tyyppi, merkki, sekä ikä, internet-yhteyden tyyppi, omistavatko he mikrofonin ja minikäläisiin arkipäiväisiin toimiin he käyttivät tietokonetta. Näiden tietojen pohjalta muodostui kuvan 9 mukainen ”Hakemukset”-taulu.

Applications			
ID	Integer(4) PK	allergy	Varchar(255)
issue_date	Timestamp	diet	Varchar(255)
not_accepted	Bool	activity_assisted	Varchar(255)
accepted	Bool	tool	Varchar(255)
queue	Bool	employment	Varchar(255)
cancelled	Bool	job	Varchar(255)
valid	Bool	course_heard	Varchar(255)
state_reason	Varchar(255)	additional_info	Varchar(255)
doctor	Varchar(255)	relationship	Text
diagnosed_disease	Varchar(255)	relative_job	Bool
disease_specifier	Varchar(255)	relative_diseases	Varchar(255)
first_symptoms	Date	relative_diet	Varchar(255)
diagnosed_year	Integer(4)	relative_performance	Varchar(255)
diagnosed_month	Integer(2)	relative_goals	Varchar(255)
disease_impact	Varchar(255)	device_type	Varchar(255)
other_diseases	Varchar(255)	device_brand	Varchar(255)
hbacteria_name	Varchar(255)	device_age	Varchar(255)
hbacteria_year	Integer(4)	broadband	Boolean
objective	Varchar(255)	wifi	Boolean
completed_course_name	Varchar(255)	mic	Boolean
completed_course_date	Date	device_used	Varchar(255)
completed_course_location	Varchar(255)		
applicant_comment	Text		
applicant_comment_date	Timestamp		

Kuva 9. RS-mallin mukaisesti luotu ”Hakemukset”-taulu ja sen sarakkeet.

”Läheisten hakemukset” -tauluun oli tarkoitus tallentaa osittain samoja sarakkeita mitä ”Hakijat”-tauluun. Tauluun lisättiin samanlaisesti yksilöivä ”ID”-sarake, hakemuksen luomispäivä, hakemuksen tilalle varatut sarakkeet, sekä tilan selitykselle varattu sarake. Läheisille kohdistettujen kurssien hakemuksessa kysyttiin läheisen henkilötietojen lisäksi kuntoutuvan sairauksiin kohdistuvia tietoja. Kuntoutujalla tarkoitetaan läheisten hakemuksissa henkilöä, jonka takia läheinen hakeutuu kurssille. Heiltä koettiin tarpeelliseksi kysyä kyseisen kuntoutujan sairauden nimeä, diagnoosin vuotta ja kuukautta, sairauden vaikutusta kuntoutujan elämään ja muita hänen sairauksiaan. Näiden lisäksi läheiseltä nähtiin tarpeen kysyä hänen omia sairauksiaan, mahdollisia apuvälineitä, avuntarvetta, sekä saako hän omaishoidolta tukea. Lopuksi hakemuksessa kysytään mistä läheinen

on saanut kuulla kurssista. Näiden sarakkeiden lisäämisestä syntyi kuvan 10 mukainen taulu.

Relatives_applications	
ID	Integer(4) PK
issue_date	Timestamp
not_accepted	Bool
accepted	Bool
queue	Bool
cancelled	Bool
valid	Bool
state_reason	Varchar(255)
pat_disease	Varchar(255)
pat_disease_year	Varchar(255)
pat_disease_month	Varchar(255)
pat_other_diseases	Varchar(255)
pat_disease_affect	Varchar(255)
caregiver	Bool
own_disease	Varchar(255)
tool	Varchar(255)
performance	Varchar(255)
course_heard	Varchar(255)

Kuva 10. RS-mallin mukaisesti luotu "Läheisten hakemukset" -taulu ja sen sarakkeet.

Kurssit-taulu

"Kurssit"-taulu luotiin tallentamaan kursseihin liittyvää tietoa. Tämäkin taulu aloitettiin luomalla ensimmäiseksi "ID"-sarake. Heti sen jälkeen tauluun listattiin kurssin nimi, kurssin ajankohta, sekä sijainti. Kurssista haluttiin myös tallentaa sen sisältö ja tavoitteet, joita pyritään saavuttamaan, sekä suositukset, kenelle kurssit sopisivat. Kurseilla oli myös henkilökuntaa, joista täytyi tallentaa nimi ja heidän titteli eli ammattinimike. Nämä tiedot kirjataan tietokantaan manuaalisesti kurssia luodessaan. Kuvassa 11 on kuva edellä luodusta taulusta.

Courses	
ID	Integer(4) (PK)
course_name	Varchar(255)
date	Date
location	Varchar(255)
contents	Text
goal	Varchar(255)
recommendation	Text
staff_name	Varchar(40)
staff_title	Varchar(40)

Kuva 11. RS-mallin mukaisesti luotu Kurssit-taulu ja sen sarakkeet.

Käyttäjät-taulu

Tulevalla hakijarekisterin käyttöliittymällä tulee olemaan käyttäjiä ja heillä omat tilinsä, joilla voivat kirjautua järjestelmään. Näistä käyttäjistä täytyi luoda oma taulu, jonne tallentaa käyttäjän käyttäjänimi, joka olisi mahdollisesti sähköpostiosoite. Tämän lisäksi myös salasanalle luotiin oma sarake, joka tulee olemaan kryptattu eli salattu. Kuvassa 12 on "Käyttäjät"-taulu.

Users	
ID	Integer(4) PK
email	VarChar(255)
password	VarChar(40)

Kuva 12. RS-mallin mukaisesti luotu "Käyttäjät"-taulu ja sen sarakkeet.

Ensietopäivähakemus- ja maksusitoumus-taulu

Kaikkien taululistan mukaisten taulujen luomisen jälkeen nousi tarve luoda muutama taulu lisää. Kurssien kartoituksessa esille nousi kahdesti vuodessa pidettävä ensietopäiväkurssi. Näihin kursseihin haetaan erilaisella kurssilomakkeella ja kurssin hakemuksessa pyydettyjä tietoja oli paljon vähemmän kuin muiden kurssien hakemuksissa. Tästä johtuen näille kursseille tehdyt hakemukset täytyi tallentaa erilliseen tauluun. Tähän hakemustauluun tallennettiin samat sarakkeet ID:stä "hakemuksen tilan syy" -sarakeeseen asti niin kuin aikaisemmissa hakemustauluissa. Näiden lisäksi tarpeen oli tallentaa

diagnosoitu sairaus, allergiat, ruokavalio ja selostus avuntarpeista. Tähän kurssiin oli hakijan myös mahdollista lisätä läheisiä ja heiltä vaadittiin allergiat, ruokavaliot ja selostus avuntarpeista.

Ensietopäiväkurssista teki erikoisen myös mahdollisuus hakea maksusitoumusta. Siinä hakija saa tukea kurssin kurssimaksun maksamiseen. Tästä maksusitoumuksesta täytyi tallentaa tietokantaan tietoa. Tiedot koostuivat "ID"-sarakkeen lisäksi maksusitoumuksen antajan "nimi"- ja "osoite"-sarakkeesta, "sitoumuksen summan määrä" -sarakkeesta, sen "myöntämispäivämäärä"-sarakkeesta, sekä sarakkeesta, johon tallennetaan onko maksusitoumuksesta tullut päätös vai ei. Nämä taulut näkyvät kuvassa 13.

Obligations		Initinfo_applications	
ID	Integer(4)(PK)	ID	Integer(4) PK
issuer	Varchar(40)	issue_date	Date
address	Varchar(40)	accepted	Bool
sum	Integer(4)	queue	Bool
date	Date	cancelled	Bool
issued	Bool	valid	Bool
		state_reason	Varchar(255)
		diagnosed_disease	Varchar(255)
		allergy	Varchar(255)
		diet	Varchar(255)
		need_of_help	Varchar(255)
		relative_allergy	Varchar(255)
		relative_diet	Varchar(255)
		relative_performance	Varchar(255)

Kuva 13. RS-mallin mukaisesti luotu "Maksusitoumukset"- ja "Ensietopäivähakemukset"-taulut ja niiden sarakkeet.

Lokitaulut

Aikaisemmin mainittiin, että GDPR mukaan henkilötietojen omistajalla on oikeus tietää kuka hänen henkilötietojaan on käsitellyt. Tämän tietoturvaan liittyvän kohdan täyttämiseksi luotiin alustavasti kaksi taulua. Näihin kahteen tauluun tallennettiin käyttäjät, jotka olivat toiminnoillaan tehneet SQL-kyselyitä hakijoiden ja heidän läheistensä tietoja sisältäviin tauluihin. Tauluun tallennettiin "Käyttäjät"-taulusta ID, joka vastasi haun tehneeseen käyttäjään. Tämän lisäksi tallennettiin "Hakijat"- tai "Läheiset"-taulusta ID, joka

vastaa tehdyn haun kohdetta. Lopuksi tallennettiin haun ajankohta. Näin syntyi kaksi lokitaulua, jotka näkyvät kuvassa 14.

View_logs_applicants	
ID	Integer(10) PK
applicant_ID	Integer(4) FK
user_ID	Integer(4) FK
created_at	Timestamp

View_logs_relatives	
ID	Integer(10) PK
user_ID	Integer(4) FK
relative_ID	Integer(4) FK
created_at	Timestamp

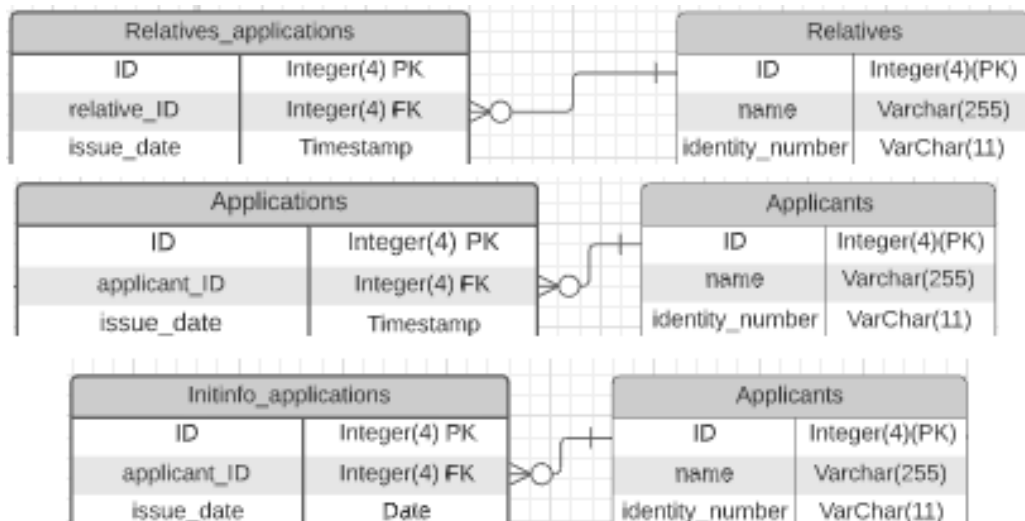
Kuva 14. RS-mallin mukaisesti luodut lokitaulut ja niiden sarakkeet.

4.10 Taulujen suhteet

Relaatiotietokantaan luotujen taulujen välille luodaan suhteita. Suhde luodaan käyttäen apuna tauluihin määriteltyjä pää- ja viiteavaimia, jotka liittävät taulut loogisesti toisiinsa. Havainnollistavana esimerkkinä voi toimia aikaisemmin luodut "Hakemukset"- ja "Kurs-sit"-taulut. Näillä tauluilla on looginen yhteys, sillä tietyllä hakemuksella haetaan johonkin tiettyyn kurssiin. Tästä syystä näiden taulujen välillä on looginen yhteys ja niiden välille on luotava suhde. Suhteiden avulla pystytään myös parantamaan taulurakenteita ja pienentämään ylimääräisen datan tallentamista tietokantaan (Hernandez, 2000). Taulujen väliset suhteet mahdollistavat myös datan tehokkaan haun useista tauluista saman aikaisesti (Hernandez, 2000). Tässä projektissa käytetyssä RS-kaaviossa taulujen välisiä suhteita havainnollistetaan käyttäen aikaisemmin kuvattuja kuvan 5 mukaisia suhdeviivoja.

Aikaisemmin luotiin kolme erilaista hakemustaulua: Kuntoutujien hakemukset, kuntoutu-jien hakemukset ensitietopäivään ja läheisten hakemukset. Näihin tauluihin tallennettiin tietoa hakijan tai heidän läheistensä tekemistä hakemuksista johonkin tiettyyn kurssiin. Tästä voitiin loogisesti päätellä, että jokainen hakemus liittyy johonkin hakijaan tai lähei-seen, riippuen mistä hakemuksesta oli kyse.

Kuntoutuvien tekemät hakemukset liittyivät loogisesti "Hakijat"-tauluun, joten "Hakemukset"-taulun ja "Hakijat"-taulun välille luotiin yhteys. Yhteydeksi asetettiin yhden suhde moneen, koska yksi hakija voi liittyä moneen "Hakemukset"-taulussa olevaan riviin ja toisinpäin yksi "Hakemukset"-taulun rivi voi liittyä vain yhteen hakijaan. Vastaavanlainen yhden suhde moneen -yhteys muodostui sekä "Hakijat"- ja "Ensitetopäivähakemukset"-taulun että "Läheiset"- ja "Läheisten hakemukset" -taulun välille. Tauluihin täytyi lisätä viiteavaimet, jotta edellä muodostetut yhteydet toimivat. Viiteavaimen lisäyksessä tärkeää on, että se vastaa alkuperäisen sarakkeen tietotyyppiä, jotta dataelementtien eheys pysyy kunnossa ja tiedot ovat verrattavissa keskenään. "Hakemukset"- ja "Ensitetopäivähakemukset"-tauluihin lisättiin "Hakijat"-taulusta "ID"-sarake toimimaan viiteavaimena. Vastaavasti läheisten "Hakemukset"-tauluun lisättiin "Läheiset"-taulusta "ID"-sarake. Nämä luodut yhteydet ovat koottu kuvaan 15.

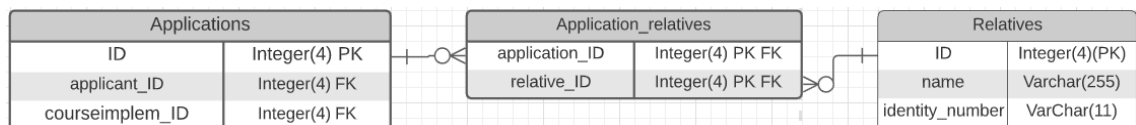


Kuva 15. Yhteydet ja viiteavaimet hakemustaulujen ja hakijoiden välillä.

Hakemukset kohdistuivat myös tiettyyn kurssiin. Tästä syystä "Hakemukset"-, "Ensitetopäivähakemukset"- ja "Läheisten hakemukset" -tauluihin lisättiin myös "Kurssit"-taulun ID viiteavaimeksi. Yhteydeksi "Kurssit"-taulun ja "Hakemukset"-, "Ensitetopäivähakemukset"- ja "Läheisten hakemukset" -taulujen välille muodostettiin myös yhden suhde moneen, koska yksi "Kurssit"-taulun rivi voi liittyä moneen hakemustaulujen riveihin ja yksi hakemustaulujen rivi voi liittyä vain yhteen "Kurssit"-taulun riviin.

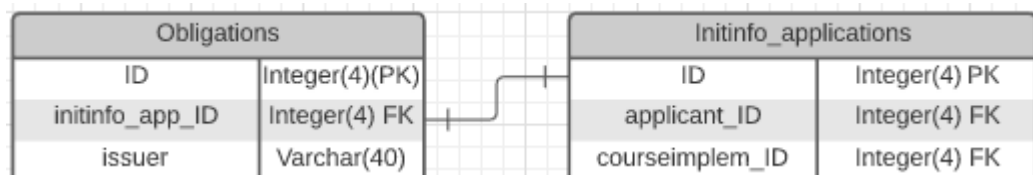
Kuntoutuvien hakijoiden ja heidän läheistensä taulut täytyi liittää toisiinsa, sillä he liittyivät loogisesti toisiinsa hakemuksien perusteella. Kuntoutuvalla hakijalla oli mahdollisuus li-

sätä monta läheistä luodessaan hakemusta. Tästä syystä järkevintä oli liittää "Hakemukset"-taulu ja "Läheiset"-taulu keskenään. Yksi läheinen pystyi olemaan monelle kuntoutuvalla hakijalle läheinen. Näiden perusteella "Hakemukset"-taulun ja "Läheiset"-taulun väliin luotiin monen suhde moneen -yhteys. Tämä yhteys täytyi kuitenkin rikkoa kahteen yhden suhde moneen -yhteyteen linkitystaulun avulla. Linkitystaulu muodostaa uuden taulun, johon tallennetaan molempien taulujen pääavaimet (Hernandez, 2000). Tämän jälkeen luodaan yhden suhde moneen -yhteydet sekä "Hakemukset"-taulun ja uuden linkitystaulun välille että "Läheiset"-taulun ja linkitystaulun välille. Tämä edellä mainittu yhteysrakenne luotiin myös "Ensitetopäivähakemukset"-taulun ja "Läheiset"-taulun välille, koska niiden välillä on myös monen suhde moneen -yhteys. Kuvassa 16 on edellä mainittu yhteysrakenne.



Kuva 16. Monen suhde moneen -yhteyden muodostaminen yhteystaulun avulla "Hakemukset"- ja "Läheiset"-taulun välille.

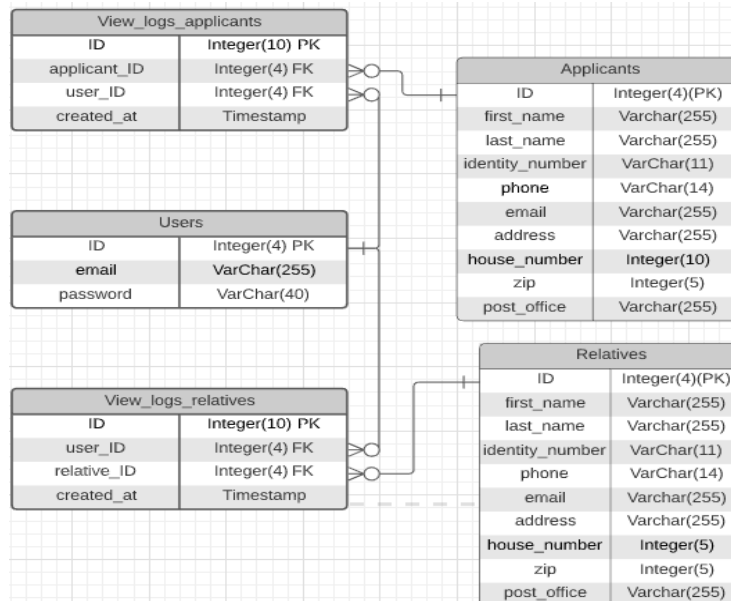
"Ensitetopäivähakemukset"-taulun ja "Maksusitoumukset"-taulun välille syntyi suhde, sillä maksusitoumukset liittyvän ensitetopäiväkurssille tehtyihin hakemuksiin. Yksi maksusitoumus liittyi vain yhteen hakemukseen ja yksi hakemus liittyi vain yhteen maksusitoumukseen. Tästä voitiin helposti päätellä taulujen välille yhden suhde yhteen -yhteys. "Maksusitoumukset"-tauluun lisättiin myös "Ensitetopäivähakemukset"-taulun ID viiteavaimeksi. Kuvassa 17 on kuvattu edellä mainittu suhde.



Kuva 17. "Maksusitoumukset"-taulun ja "Ensitetopäivähakemukset"-taulun välinen

Kuntoutuvien hakijoiden ja läheisten tietojen tarkastelusta täytyi jäädä merkintä tietokannan lokitauluihin. "Hakijat"-tauluun kohdistuneet kyselyt tallentuivat "Hakijat loki"-tauluun

ja ”Läheiset”-tauluun kohdistuneet kyselyt ”Läheiset loki” -tauluun. Näiden välille luotiin yhden suhde moneen -yhteys, koska yksi rivi lokitaulussa kohdistuu yhteen hakijaan tai läheiseen ja yksi läheinen tai hakija voi kohdistua moneen lokitaulun riviin. Lopuksi ”Käyttäjät”-taulu liitettiin yhden suhde moneen -yhteydellä molempiin lokitauluihin, koska yksi käyttäjä voi liittyä moneen lokitapahtumaan ja yksi lokitapahtuma voi liittyä vain yhteen käyttäjään. Kuvassa 18 näkyy edellä luodut yhteydet.



Kuva 18. Yhden suhde moneen -yhteydet loki-, ”Käyttäjät”-, ”Hakijat”- ja ”Läheiset”-taulujen välillä.

4.11 Normalisointi

Tietokannan taulujen normalisointi on tärkeä prosessi suunnittelun aikana. Normalisointiprosessin tavoitteena on parantaa tietokannan tallennusmuotoa pienentämällä datan toistuvuutta tietokannassa, vähentämällä datan päivitykseen liittyviä ongelmia ja mahdollistaakseen tulevien rakennemuutoksien joustavuutta (Hovi ym. 2005). E. F. Codd:in mukaan prosessin aikana tietokannan skeema testataan ja varmistetaan, että se täyttää halutun normaalimuodon ehdot. Hän ehdotti noin vuonna 1972 normaalimuotoja olevan kolme eri tasoa, joita tässä projektissa käytettiin tietokantarakenteen skeeman viimeistelemiseksi. Nämä muodot ovat ensimmäinen normaalimuoto (1NF), toinen normaalimuoto (2NF) ja kolmas normaalimuoto (3NF) (Elmasri & Navathe 2007).

Ensimmäinen normaalimuoto (1NF)

Jokaisella tasolla on omat säännöt, joita normalisointiprosessissa noudatetaan. Tietokannan täytyy olla normalisoitu edelliseen muotoon, jotta se voi edetä seuraavalle tasolle. Normalisointi aloitetaan muuttamalla luotu tietorakenneskeema 1NF:ään. 1NF:ssä tavoitteena on hajottaa tietokannan tauluista sarakkeet, joihin tallennetaan toistuvia arvoja tai moniarvoisia dataelementtejä, sekä ehkäistään tyhjiä dataelementtien muodostumista. Ne hajotetaan useaan eri sarakkeeseen tai kokonaan uuteen tauluun. (Elmasri & Navathe 2007.) (Hovi ym. 2005.)

Ensimmäisen tason normalisoinnissa käytiin kaikki tietokantaskeemaan luodut taulut ja niiden sisältämät sarakkeet läpi yksitellen. "Hakijat"- ja "Läheiset"-taulusta "nimi"- ja "osoite"-sarakkeet täytyi hajottaa useaan sarakkeeseen. "Nimi"-sarake hajotettiin "etu"- ja "sukunimi"-sarakkeiksi, koska muuten nimisarakeeseen tallennettaisiin moniarvoista dataa. Samasta syystä "osoite"-sarake hajotettiin "lähiosoite"-, "talon numero"-, "postinumero"- ja "postitoimipaikka"-sarakkeiksi. Muihin sarakkeisiin ei tarvinnut koskea, koska ne täyttivät 1NF mukaiset säännöt. Kuvassa 19 näkyy 1NF normalisoidut "Hakijat"- ja "Läheiset"-taulut.

Applicants		Relatives	
ID	Integer(4)(PK)	ID	Integer(4)(PK)
first_name	Varchar(255)	first_name	Varchar(255)
last_name	Varchar(255)	last_name	Varchar(255)
identity_number	VarChar(11)	identity_number	VarChar(11)
phone	VarChar(14)	phone	VarChar(14)
email	Varchar(255)	email	Varchar(255)
address	Varchar(255)	address	Varchar(255)
house_number	Integer(10)	house_number	Integer(5)
zip	Integer(5)	zip	Integer(5)
post_office	Varchar(255)	post_office	Varchar(255)

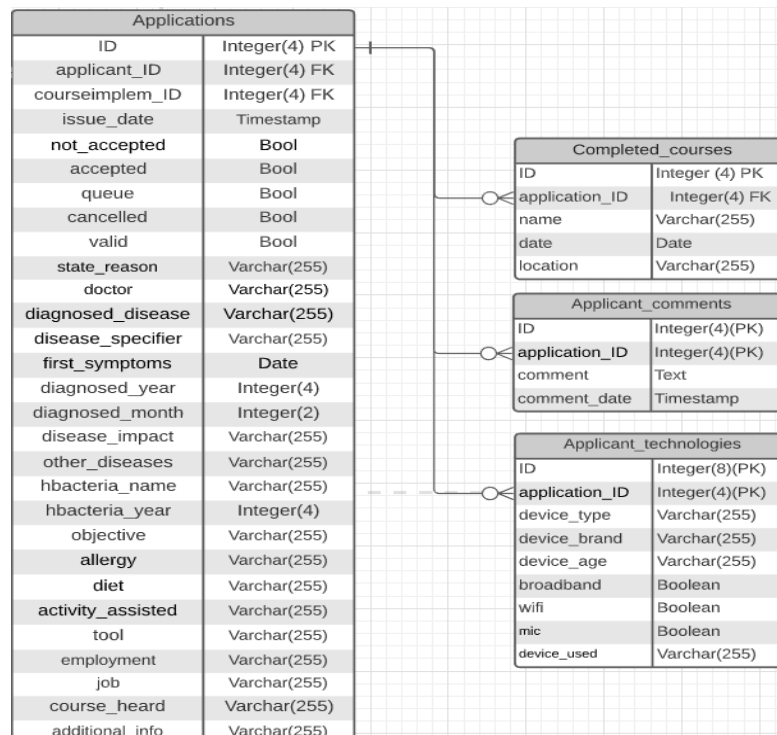
Kuva 19. "Hakijat"- ja "Läheiset"-taulut normalisoitu 1NF

Seuraavaksi oli hakemustaulujen vuoro. Kuntoutuvien hakijoiden "Hakemukset"-tauluun tallennetaan paljon hakijoiden hakemuksiin kirjoittamia vapaamittaisia tekstejä, kuten sairautensa vaikutusta elämäänsä, muiden sairauksien luettelu ja päivittäiset avuntarpeet. Tästä syystä sarakkeiden muuttaminen ei ollut aivan selkeää. Ensimmäisenä muutosta vaativat sarakkeet, jotka liittyivät aikaisemmin osallistuttuihin kursseihin. Hakijoilla ei välttämättä ole aikaisempaa osallistumishistoriaa kursseilta, joka aiheuttaisi näiden sarakkeiden päätyvän tyhjäksi kyseisellä rivillä. Tämän toistuesssa tauluun muodostuisi

paljon tyhjiä dataelementtejä näihin sarakkeisiin. Tämän ehkäisemiseksi "Hakijat"-taulusta poistettiin "aikaisemman kurssin nimi"-, "ajankohta"- ja "sijainti"-sarakkeet, ja niille tehtiin oma taulu. Taulun nimeksi tuli suoritettut kurssit, joka kuvastaa hyvin sinne tallennettavia tietoja aikaisemmista kursseista. Tauluun lisättiin viiteavaimeksi "Hakemukset"-taulun ID ja yhden suhde moneen -yhteys, koska yhteen hakemukseen voi kirjata monta aikaisemmin suoritettua kurssia ja yksi aikaisemmin suoritettu kurssin tallenne viittaa yhteen hakemukseen.

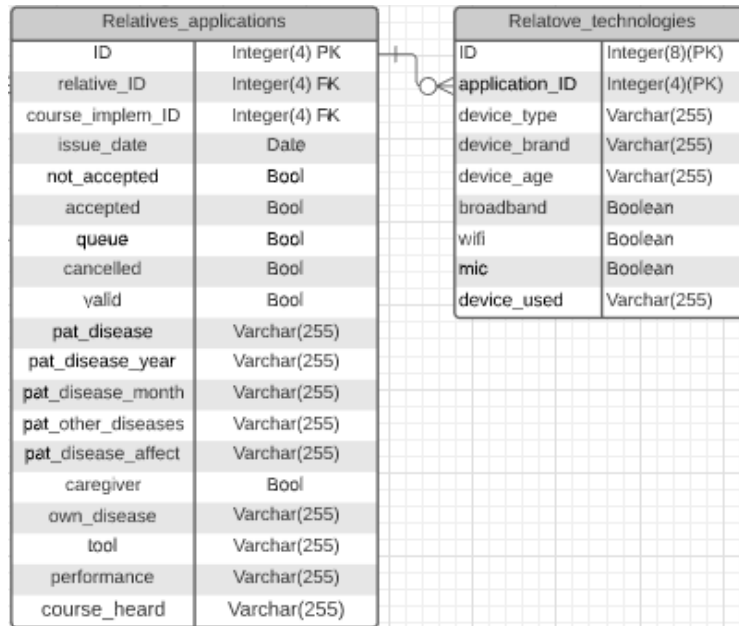
Seuraavaksi tuli vuoroon sarakkeet, joihin oli tarkoitus tallentaa parkinsonliiton työntekijöiden kirjaamia muistiinpanoja ja kommentteja, sekä niiden kirjausajankohdat. Näillä sarakkeilla oli kolme ongelmaa nykyisellä paikallaan. Ensimmäisenä oli niiden mahdollisuus jäädä tyhjäksi. Kirjatut muistiinpanot ja kommentit tullaan tallentamaan kurssin aikana ja jälkeen tai niitä ei kirjata ollenkaan. Tämä johtaisi moneen tyhjään dataelementtiin kyseisillä sarakkeilla. Toinen ja kolmas ongelma kohdistuu tiedon tallentamiseen. Kommentteja oli mahdollisuus kirjata useampia, joten kyseiseen dataelementtiin tallennettaisiin monta kommenttia eli siitä tulisi moniarvoinen. Toinen mahdollisuus olisi kirjata aina uusi rivi kommentteille, mutta silloin tulisi turhaan samanlaisia hakemusriviä. Näistä syistä "kommentti"- ja sen "kirjauspäivä"-sarakkeet siirrettiin omaan tauluun. Tähän tauluun liitettiin myös viiteavaimeksi "Hakemukset"-taulun ID ja yhden suhde moneen -yhteys.

Seuraavaksi "Hakemukset"-taulussa tuli vastaan sarakkeet, joihin tallennettiin tietoa hakijan tietoteknisistä laitteista, kuten tietokoneesta ja internet-yhteydestä. Näitä tietoja kerättiin, jos hakija osallistui verkkokurssille. Tästä syystä nämä tiedot siirrettiin myös omaan tauluun, jotta ne eivät loisi turhaan tyhjiä sarakkeita tilanteissa, joissa hakija ei hakeutuisi verkkokurssille. "Hakijan tietotekniikat"-taulu luotiin ja sinne siirrettiin laitteen "tyyppi"-, "merkki"- ja "ikä"-sarake, sekä laajakaistan, wifin ja mikrofonin totuusarvot sarakeet. Viimeisenä siirrettävänä oli sarake, johon tallennettiin tietoa tietoteknisten laitteiden käyttökohteesta. Myös tähän tauluun lisättiin viiteavaimeksi "Hakemukset"-taulun ID ja yhden suhde moneen -yhteys. Kuvassa 20 on "Hakemukset"-taulun 1NF normalisoinnin tulos



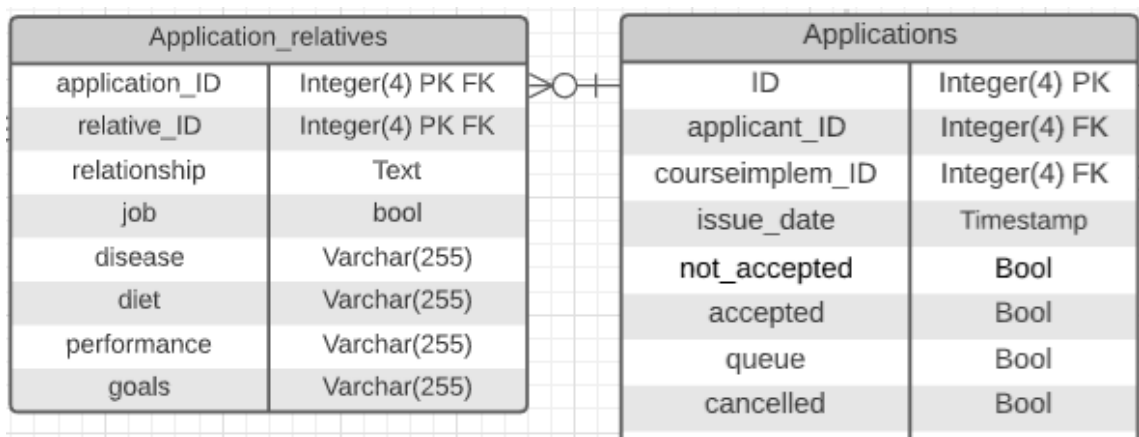
Kuva 20. 1NF mukaan normalisoitu "Hakemukset"-taulu ja uudet taulut.

"Hakemukset"-tauluun tallennettavat tiedot heidän läheisistä täytyi myös siirtää. Hakijoilla oli mahdollisuus kirjata useita läheisiä tai ei ollenkaan. Näiden sarakkeiden jättäminen "Hakemukset"-tauluun aiheuttaisi tyhjiä dataelementtejä riveille niiden sarakkeiden kohdalle, jos läheisiä ei kirjattaisi. Päinvastaisesti tauluun tulisi toistuvaa dataa tai dataelementeistä tulisi moniarvoisia, jos läheisiä kirjattaisiin enemmän kuin yksi. Uuden taulun sijaan, tiedot siirrettiin "Hakemukset"-taulun ja "Läheiset"-taulun välille luotuaan yhteystauluun. Taulun tarkoituksena oli tallentaa yhdelle riville yhden läheisen ja yhden hakemuksen välinen yhteys, joten oli loogista siirtää läheisten "työ"-, "sairaus"-, "ruokavalio"-, "toimintakyky"- ja "tavoite"-sarakkeet tähän tauluun. "Ensitetopäivähakemukset"-taulusta siirrettiin samanlailla läheisiin liittyvät sarakkeet yhteystauluun, joka liittää "Läheiset"-taulun ja "Ensitetopäivähakemukset"-taulun. Kuvassa 21 on tämän normalisoinnin tulos.



Kuva 21. "Läheisten hakemukset" -taulu 1NF normalisoinnin jälkeen.

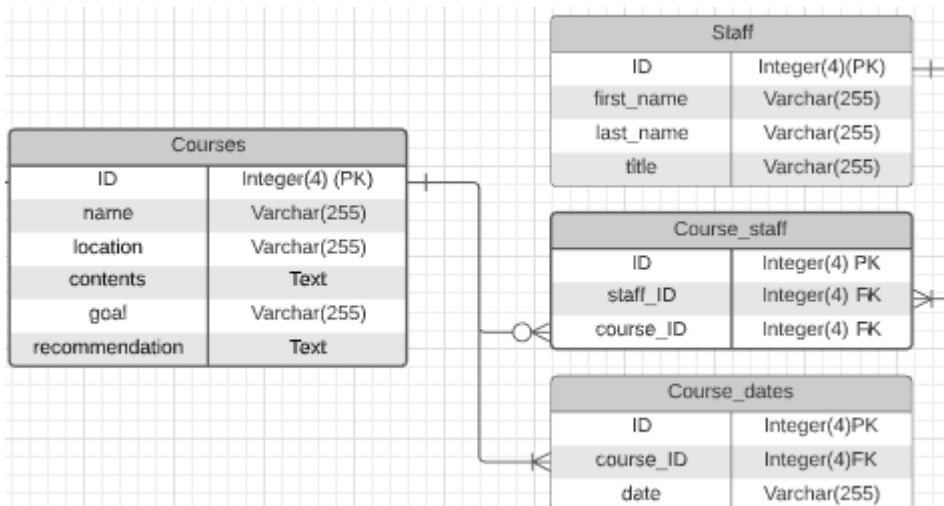
Seuraavana oli "Läheisten hakemukset" -taulu. Siinä pidettiin samaa periaatetta, jota pidettiin hakijoiden "Hakemukset"-taulua normalisoitaessa. Sarakkeita käytiin läpi yksitellen. Ainoana 1NF säännön vastaisena pidettiin sarakkeita, joiden tarkoitus oli tallentaa dataa läheisten tietoteknisistä laitteista, jos hakemuksessa haettiin verkkokurssille. Normaalille kurssille ilmoittautuessa kyseiset sarakkeet jäisivät tyhjäksi. Tiedot siirrettiin omaan tauluun, jonka nimeksi tuli läheisten tietotekniikat. Tauluun liitettiin myös viitevaimeksi "Läheisten hakemukset" -taulun ID ja suhteeksi yhden suhde moneen. Kuvasssa 22 on tämän normalisoinnin tulos.



Kuva 22. "Hakemuksien läheiset" -taulu "Hakemukset"-taulun 1NF normalisoinnin jälkeen.

Seuraavana oli vuorossa "Kurssit"-taulu. Kurssin nimi ja sijainti pysyi pääsääntöisesti samana, mutta kurssin ajankohtia oli monta. Tämän sarakkeen säilyttäminen kurssit taulussa aiheuttaisi moniarvoisen dataelementin syntymisen. "Kurssin ajankohta" -sarake täytyi siirtää omaan tauluun, joka pitäisi kirjata eri kurssien ajankohdista. Taulun nimeksi tuli "Kurssin ajankohdat". Tähän tauluun liitettiin viiteavaimeksi "ID"-sarake "Kurssit"-taulusta ja taulujen suhteeksi yhden suhde moneen -yhteys.

Kurssin henkilökunnan nimien ja arvonimien tallentamiseen tarkoitettavat sarakkeet täytyi myös siirtää omaan tauluunsa. Niiden jättäminen kurssit tauluun aiheuttaisi myös moniarvoiset dataelementit tai pakottaisi toistuvien datarivien muodostamisen. Näille sarakkeille luotiin uusi taulu nimeltä "Henkilökunta" ja siihen lisättiin viiteavaimeksi "Kurssit"-taulun ID. "Nimi"-sarake jaettiin samalla "etunimi"- ja "sukunimi"-sarakkeiksi. Suhteeksi muodostui monen suhde moneen, koska kurseilla voi olla monta henkilökunnan jäsentä ja yksi henkilökunnan jäsen voi liittyä moneen eri kurssiin. Tämän suhteen takia taulujen välille täytyi luoda yhteystaulu. Yhteystaulun nimeksi tuli "Kurssin henkilökunta" ja siihen tallennettiin "Kurssit"-taulun ja "Henkilökunta"-taulun ID avaimet. Yhteystaulun ansiosta sekä "Kurssit"- että "Henkilökunta"-taulusta muodostettiin yhden suhde moneen -yhteydet "Kurssin henkilökunta" -tauluun. Kuvassa 23 on tämän normalisoinnin tulos.

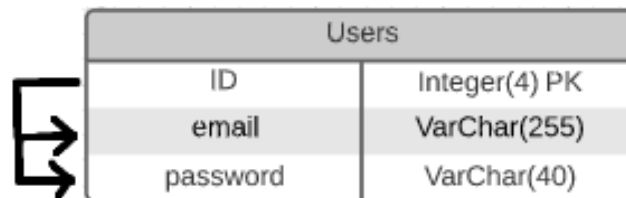


Kuva 23. "Kurssit"-taulun normalisointi 1NF muotoon.

Jäljelle jääneet taulut käytiin myös läpi yksitellen. "Käyttäjät"- ja lokitauluissa ei ollut mitään normalisoitavaa, joten ne annettiin olla nykyisellään. Viimeisestä käydystä "Maksusitoumukset"-taulusta löytyi 1NF mukaista normalisoitavaa. Taulusta löytyi moniarvoinen "osoite"-sarake, joka täytyi jakaa useaan sarakkeeseen. Uudet sarakkeet olivat "lähiosoite", "talonnumero", "postinumero" ja "postitoimipaikka".

Toinen normaalimuoto (2NF)

Toiseen normaalimuotoon (2NF) voidaan siirtyä, kun kaikki tietokannan taulut on normalisoitu 1NF:n mukaisesti. 2NF:ssä tarkoitetaan tarkastaa taulujen sisältämien sarakkeiden funktionaalisia riippuvuuksia ja pyrkiä poistamaan osittaiset funktionaaliset riippuvuudet (Elmasri & Navathe, 2007). Funktionaalisella riippuvuudella tarkoitetaan taulun sarakkeiden riippuvuutta pääavaimesta. Esimerkiksi kuvassa 24 on "Käyttäjät"-taulu. Siinä funktionaalista riippuvuutta kuvataan nuolilla. "Sähköposti"- ja "salasana"-sarakeet ovat funktionaalisesti riippuvaisia "ID"-sarakeesta, koska yhdellä ID:llä on vain yksi sähköposti ja salasana. Taulut, joissa pääavaimena toimii yksi sarake, voidaan jättää 2NF testaamisen ulkopuolelle niiden jo ollessa 2NF muodossa. Suurin osa tämän projektin tietokannan skeemasta koostuu näistä yhden pääavaimen tauluista.



Kuva 24. "Käyttäjät"-taulu ja sen sarakkeet.

Osittaisella funktionaalisella riippuvuudella tarkoitetaan tilannetta, jossa taululla on enemmän kuin yksi sarake, jotka muodostavat yhdessä pääavaimen ja normaali sarake ei ole riippuvainen kaikista sarakkeista vaan vain osasta. Esimerkiksi "Hakemukset"-taulussa olisi osittainen funktionaalinen riippuvuus, jos tauluun tallennettaisiin hakijan nimi. Tällöin nimisarake olisi riippuvainen hakemuksen "ID"- ja hakijan "ID"-sarakeista, mutta ei kurssin "ID"-sarakeesta. Näitä tilanteita ei kuitenkaan löytynyt mistään taulusta, joten tauluihin ei tehty muutoksia ja 2NF normalisointi oli valmis.

Kolmas normaalimuoto (3NF)

Viimeisessä normalisoinnin vaiheessa eli 3NF:ssä etsitään tauluista sarakkeita, jotka ovat funktionaalisesti riippuvaisia pääavaimesta, mutta samalla myös jostain tavallisesta sarakkeesta. Tätä kutsutaan myös transitiiviseksi riippuvuudeksi. (Elmasri & Navathe 2007)

"Hakijat"- ja "Läheiset"-taulussa on hyvin yleisesti esitetty esimerkki transitiivisesta riippuvuudesta. Näiden taulujen "postinumero"- ja "postitoimipaikka"-sarakkeet ovat funktionaalisesti riippuvaisia taulujen "ID"-sarakkeesta, mutta "postitoimipaikka"-sarake on samalla funktionaalisesti riippuvainen "postinumero"-sarakkeesta. 3NF mukaisesti "postitoimipaikka"-sarake täytyisi siirtää omaan tauluunsa, mutta tietoturvan takia sitä ei siirretty. "Hakijat"- ja "Läheiset"-taulu ja sen sisältämä data tullaan tietokannan toteutusvaiheessa salaamaan. Uuden taulun luominen loisi ylimääräisen taulun, joka täytyisi salata ja se aiheuttaisi myös ylimääräisten hakujen luomisen.

"Hakemukset"-taulussa transitiivisia riippuvuuksia löytyi monta, kuten diagnosoituun sairauteen liittyvät sarakkeet. Näitä oli "tarkenne"-, "diagnoosin kuukausi" -, "diagnoosin vuosi" -, "ensioireet"- ja "sairauden vaikutukset" -sarake. Samat sarakkeet olivat myös "Läheisten hakemukset" -taulussa, lukuun ottamatta "tarkenne"- ja "ensioireet"-sarakkeita. Nämä olisi täytynyt siirtää hakemustauluista omiin tauluihin, mutta tietoturvan ja tietokantakyselyiden minimoimisen takia ne päätettiin jättää hakemustauluihin. Näin syntyi 3NF:n mukaisesti normalisoitu tietokannan ER-kaavio (Liite 2).

5. YHTEENVETO

Tämä opinnäytetyö oli osa projektia, jossa päämääränä oli suunnitella, toteuttaa ja testata kuntoutuskurssien hakijarekisterin tietokanta ja käyttöliittymä, sekä kurssien ilmoittautumislomake. Opinnäytetyössä kuitenkin keskityttiin projektin suunnitteluvaiheeseen ja tietokannan rakenteen suunnitteluun.

Opinnäytetyön päätavoitteena oli suunnitella tietokanta kuntoutuskurssien hakijarekisterille ja dokumentoida sen rakenne niin, että se voidaan myöhemmin toteuttaa. Lisäksi tietokannasta täytyi suunnitella mahdollisimman selkeä, turvallinen ja toimeksiantajan vaatimuksien ja toiveiden mukainen. Pääosin näihin tavoitteisiin päästiin, vaikka opinnäytetyöhön liittyvän projektin alustavaan aikatauluun ei ehditty ja opinnäytetyön oma aikataulu alkoi tulla vastaan.

Aikataulumuutoksista huolimatta lopputulokseksi saatiin hahmotettua tietokannan tietotarpeet ja rakenne, sekä luotua niistä ER-käsitelmä hyödyntävä visuaalinen diagrammi. Diagrammista tuli selkeä ja niin turvallinen kuin osasimme suunnitella. Se täytti myös toimeksiantajan vaatimukset ja toiveet siltä osin, mitä ilman toteutus- ja testausvaihetta voitiin todeta.

Tietokannan rakenteen suunnitteluvaiheen tärkeys korostui mitä pidemmälle siinä edettiin ja siksi projektin aikataulua päätettiin muuttaa ja antaa suunnittelulle enemmän aikaa. Tässä kohtaa päätin myös tiputtaa opinnäytetyöstä toteutusvaiheen ja keskittyä tietokannan suunnitteluun ja sen vaiheisiin. Kahtena suurimpana syynä projektin ja opinnäytetyön aikataulumuutoksiin oli mielestäni projektin hidas aloitus ja vaativuuden kasvaminen. Hidas aloitus johtui pääosin sopivan opiskelijapohjaisen projektitiimin löytämisestä ja koostamisesta. Myös heidän omat aikataulunsa ja koulun tuomat kurssit vaikuttivat aloituksen hitauteen. Tästä syystä myös opinnäytetyön aloittaminen hidastui. Vaativuuteen vaikutti projektitiimin henkilökohtainen osaaminen ja GDPR:n tuomat vaatimukset henkilötietojen käsittelyyn.

Suunnittelu alkoi GDPR:ään ja projektin tavoitteisiin tutustumalla. Vaihe oli kriittinen projektin etenemisen kannalta, sillä projekti kasvoi vaiheen aikana. Vaadittu järjestelmä ja siihen liitettävä tietokanta vaikutti aluksi yksinkertaisemmalta ja helpommalta suunnitella ja toteuttaa kuin mitä se loppujen lopuksi oli. Vaadittujen tietojen määrä kasvoi nopeasti ja tärkeää oli kartoittaa haluttu data tärkeysjärjestykseen eli mitkä olivat pakollisia ja

mitkä eivät. GDPR toi omat vaatimukset ja pakolliset toimet siitä, miten tietojenkäsittelyyn kohdistuva sovellus ja sen tietokanta tulisi suunnitella.

Itse olin ihan tyytyväinen tietokannan rakenteen lopputulokseen. Luotua rakennetta ei kuitenkaan ole näytetty tietoturvan tai tietokantojen asiantuntijalle eikä se ole saanut heiltä kommentteja tai parannusehdotuksia. Tästä syystä mielestäni rakennetta ei välttämättä sellaisenaan voida luokitella parhaaksi mahdolliseksi ratkaisuksi tämän tyyppisen tietokannan luomiseen. Mielestäni luotua tietokannan rakennetta kannattaisi näyttää tietoturvan tai tietokantojen asiantuntijalle, jotta sitä voitaisiin parantaa. Tulosta ja siihen johtaneita ratkaisuja voidaan kuitenkin hyödyntää samankaltaisissa projekteissa, kuitenkin pitämällä mielessä edellä mainittu asia.

Opinnäytetyöstä opin kuinka paljon GDPR:llä on vaikutusta nykypäivän tietojenkäsittelyyn ja kuinka GDPR:ää sovelletaan. Lisäksi opin kuinka tietokannan suunnitteluvaihe on tärkeä eheän ja toimivan tietokannan luomiseen.

Opinnäytetyössä ilmi tullut projekti mahdollisesti jatkuu opinnäytetyön jälkeen. Jos se jatkuu, niin siinä tullaan valitsemaan ohjelmointikieli, jolla hakijarekisterin käyttöliittymä aloitetaan luomaan ja relaatiotietokantajärjestelmä, jonka avulla suunniteltu tietokanta toteutetaan. Toteutusvaiheen jälkeen luotu hakijarekisteri ja tietokanta tullaan testaamaan toimivaksi ja turvalliseksi ennen niiden käyttöönottoa.

LÄHTEET

1. Ameller, D., Ayala, C., Cabot, J. & Franch, X. 2012. How do software architects consider non-functional requirements: An exploratory study. Viitattu 19.10.2020. <https://ieeexplore.ieee.org/document/6345838>.
2. Elmasri, R. & Navathe, S. B. 2007. Fundamentals of Database Systems. 5. painos. Greg Tobin.
3. Euroopan parlamentin ja neuvoston asetus (EU) 679/2016. Annettu 27.4.2016. Asetus luonnollisten henkilöiden suojelusta henkilötietojen käsittelyssä sekä näiden tietojen vapaasta liikkuvuudesta ja direktiivin 95/46/EY kumoamisesta (yleinen tietosuojasetus). Viitattu 19.10.2020 <https://eur-lex.europa.eu/legal-content/FI/TXT/?uri=CELEX%3A32016R0679>.
- 4.theFirma, 2020. theFirma. Viitattu 19.10.2020 https://www.thefirma.fi/?page_id=11.
5. Hernandez, M. J. 2000. Tietokannat: suunnittelu käytännössä. Käännöspainos. Jyväskylä: Gummerus Kirjapaino Oy.
6. Hovi, A. 2013. SQL-opas. 13. painos. Jyväskylä: Docendo Oy.
7. Hovi, A., Huotari, J. & Lahdenmäki, T. 2005. Tietokantojen suunnittelu ja indeksointi. 2. painos. Jyväskylä: Docendo Oy.
8. Juvonen, R. 2018. Ohjelmistoprojektin sudenkuopat ja miten ne vältetään. Helsinki: BoD tmi - Books on Demand.
9. Microsoft 2020. Microsoft Planner. Viitattu 19.10.2020. <https://www.microsoft.com/fi-fi/microsoft-365/business/task-management-software>.
10. Mäntyneva, M. 2016. Hallittu projekti: Jäntevästä suunnittelusta menestykselliseen toteutukseen. Helsinki: Kauppakamari.
- 11.Parkinsonliitto 2020. Parkinsonliitto. Viitattu 19.10.2020. <https://www.parkinson.fi/parkinsonliitto>.
12. Schwaber, K. & Sutherland, J. 2017. Scrum-opas: Scrumin määritelmä ja pelisäännöt. Saatavissa: <https://scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Finnish.pdf>.

13. ScrumAlliance 2020. Scrumin määritelmä. Viitattu 14.11.2020. <https://www.scrumalliance.org/about-scrum/definition>.

14. Tietosuoja laki 2018/1050 2018. Saatavissa: <https://www.finlex.fi/fi/laki/alkup/2018/20181050>.

15. Tietosuojavaltuutetun toimisto 2020. Mikä on henkilötieto?. Viitattu 19.10.2020. <https://tietosuoja.fi/mika-on-henkilotieto>.

Projektin määrittelydokumentti

Projekti	Valmennuskurssien hakijarekisterin toteutus		
Osapuolet/ yhteyshenkilöt			
Aikataulu	<p>Projekti toteutetaan 24.8.2020 – 29.1.2021 välisenä aikana kolmessa vaiheessa:</p> <p>Vaihe 1: Hakijarekisterin rakenteen suunnittelu 24.8. – 2.10.2020</p> <p>Vaihe 2: Hakijarekisterin rakentaminen ja ilmoittautumislomakkeen toteutus verkkosivuille 5.10.2020 – 8.1.2021</p> <p>Vaihe 3: Hakijarekisterin testaaminen ja bugien korjaaminen 11.1. – 29.1.2021</p>		
Projektin tavoite	<p>Projektin tavoitteena on toteuttaa hakijarekisteri Parkinsonliitto ry:n valmennuskurssseille. Valmennuskurssseille ilmoittautuminen tapahtuu Parkinsonliitto ry:n verkkosivuilla olevan kurssilistauksen kautta. Valitulle kurssille ilmoitaudutaan esitäytetyllä lomakkeella (kurssin tiedot esitäytettynä), joka vie ilmoittautuneen henkilön tiedot eteenpäin hakijarekisteriin. Hakijarekisterissä ilmoittautuneet yhdistetään automaattisesti sopiville kurssseille. Koska hakijarekisteri sisältää henkilötietoja ja terveydentilaa koskevaa informaatiota, hakijarekisteriä rakentaessa varmistetaan rekisterin tietoturvasuus ja GDPR:n mukainen toteutus. Parkinsonliitto ry:n nykyinen hakijarekisteri on rakennettu Microsoft Accessin avulla ja projektissa myös siirretään Microsoft Accessissa oleva tieto uuteen hakijarekisteriin. Henkilötietojen rekistereiden välistä siirtoa varten Parkinsonliitto ry:n (rekisterinpitäjä) ja Turun Ammattikorkeakoulun (rekisterin käsittelijä) tulee tehdä GDPR:n mukainen henkilötietojen käsittelysopimus.</p>		
Menetelmät	<p>Projekti toteutetaan projektitoimisto theFIRMAN opiskelijatyönä sen käytössä olevilla työkaluilla. Asiakas osallistuu projektin suunnitteluun yhdessä opiskelijoiden ja henkilökunnan kanssa sekä toimittaa tarvittaessa projektiin vaadittavan sisällön.</p> <p>Vaihe 1. Projektin ensimmäisessä vaiheessa suunnitellaan hakijarekisterin rakennetta ja selvitetään sopivimpia työkaluja rekisterin rakentamiseen.</p>		

Vaihe 2. Projektin toisessa vaiheessa rakennetaan hakijarekisteri ja Parkinsonliitto ry:n sivuille asetettava ilmoittautumislomake.

Vaihe 3. Projektin kolmannessa vaiheessa testataan hakijarekisteriä ja korjataan mahdolliset bugit. Tässä vaiheessa myös tuodaan vanhat tiedot Microsoft Accessista uuteen hakijarekisteriin.

<p>Kohteen rajaus</p>	<p><u>Sisältö:</u></p> <ul style="list-style-type: none"> - Projektissa toteutetaan hakijarekisteri Parkinsonliitto ry:n tarjoamille valmennuskursseille, joka automaattisesti yhdistää ilmoittautuneet sopiville kursseille. - Hakijarekisterin tietoturvasuus varmistetaan ja rekisteri rakennetaan GDPR:n mukaisesti. - Projektissa toteutetaan Parkinsonliitto ry:n verkkosivuille ilmoittautumislomake, jonka kautta ihmiset voivat ilmoittautua valmennuskursseille. - Vanhan hakijarekisterin tiedot tuodaan Microsoft Accessista uuteen hakijarekisteriin. <p><u>Asiakas:</u></p> <ul style="list-style-type: none"> - Osallistuu hakijarekisterin suunnitteluun projektiryhmän kanssa. - Valitsee ja kustantaa tarvittaessa ylläpitopalvelunsa. Asiakas myös toimittaa tarvittavat tunnukset projektiryhmälle hyvissä ajoin ennen online-ympäristöön pystyttämistä. - Jakaa projektiryhmälle tarvittavat tunnukset verkkosivujen muokkaamiseen ilmoittautumislomaketta varten ja Microsoft Accessiin vanhojen tietojen siirtämistä varten. <p><u>Projektiryhmä:</u></p> <ul style="list-style-type: none"> - Suunnittelee hakijarekisterin rakenteen ja selvittää sopivimman työkalun rekisterin rakentamiseen. - Rakentaa hakijarekisterin ja Parkinsonliitto ry:n sivuille asetettavan ilmoittautumislomakkeen. - Varmistaa hakijarekisterin tietoturvasuuden ja toteuttaa rekisterin GDPR:n mukaisesti. - Testaa hakijarekisteriä ja korjaa mahdolliset bugit. - Tuo vanhat tiedot Microsoft Accessista uuteen hakijarekisteriin.
----------------------------------	---

Lopullinen ER-Kaavio

