

SAVONIA

ammattikorkeakoulu

OPINNÄYTETYÖ - AMMATTIKORKEAKOULUTUTKINTO
TEKNIIKAN JA LIIKENTEEN ALA

DATAMIGRAATIOON PEREHDYTTÄMINEN, TEKNINEN TOTEUTUS JA OHJEISTUS

TEKIJÄ/T Jani Passi

Koulutusala Tekniikan ja liikenteen ala	
Tutkinto-ohjelma Tietotekniikan tutkinto-ohjelma	
Työn tekijä(t) Jani Passi	
Työn nimi Data migraatioon perehdyttäminen, tekninen toteutus ja ohjeistus	
Päiväys 27.02.2022	Sivumäärä/Liitteet 28
Toimeksiantaja/Yhteistyökumppani(t) Pinja Digital Oy	
Tiivistelmä <p>Työn toimeksiantajana toimi Pinja Digital Oy. Toimeksiantajalla oli tarve tehostaa datamigraatiotyöhön perehdyttämisen prosessia teknisen toteutuksen ohjeistuksella. Tämän tarpeen pohjalta lähdettiin toteuttamaan suunnitelmaa perehdytysoppaan toteutukselle.</p> <p>Työn teoriaosuudessa on kuvattu datamigraatiotyön prosessit projektina, selitetty keskeiset käsitteet datamigraatiotyössä ja esitelty toimeksiantajalla käytössä olevat työkalut, joilla datamigraatio voidaan toteuttaa. Työkalu, jolla tämän työn datamigraatio toteutettiin, oli JetBrainsin DataGrip -sovellus. Perehdytysoppaaseen tarvittiin lisäksi kuvitteellinen aloitusdata, joka kuvasti työssä lähtöjärjestelmän dataa. Aloitusdatan muodostaminen päätettiin toteuttaa Mockaroo.com-sivustolla, joten teoriaosuus ja toteutus sisältää myös tämän sivuston peruskäytön opastuksen.</p> <p>Toteutusvaihe aloitettiin suunnitteleamalla perehdytysoppaan rakenne ja suunnitteleamalla geneerinen aloitusdata valitulla palvelulla, joka oli helposti tallennettava ratkaisu perehdytysoppaan teknilliseen osuuteen käytettäväksi. Työssä toteutettiin datamigraatio aloitusdatan ja toimeksiantajan toiminnanohjausjärjestelmän tietokantarakenteenvälillä käyttäen toimeksiantajan käytössä olevia työkaluja ja käytäntöjä. Toteutuksen jälkeen työn prosessit kirjattiin ylös vaiheittain perehdytysoppaalle tukemaan jo olemassa olevia käytäntöjä ja toimeksiantajan materiaaleja.</p> <p>Lopputuloksena saatiin kirjattua teknillinen perehdytysopas toimeksiantajalle käyttöön tukemaan toimeksiantajan työntekijän perehdytystä datamigraatiotyöhön ja samalla opinnäytetyön tekijä perehdytti itsensä tähän työtehtävään. Perehdytysopas sisältää teoriaosuuden, teknillisen toteutuksen ja toimeksiantajan datamigraatiotyönprosessin ja käytännöt.</p>	
Avainsanat Datamigraatio	

Field of Study Technology, Communication and Transport	
Degree Programme Degree Programme in Information Technology	
Author(s) Jani Passi	
Title of Thesis Introduction to Data Migration, Technical Implementation, and Guidance	
Date 27.02.2022	Pages/Appendices 28
Client Organisation /Partners Pinja Digital Oy	
<p>Abstract</p> <p>The aim of this thesis was to streamline the process of introduction for data migration work with technical implementation guidelines. Based on this need, a plan was made to implement the introduction guide. This thesis was commissioned by Pinja Digital Oy.</p> <p>The theoretical part of the thesis described the processes of data migration work as a project. The key concepts in data migration were explained and the tools used by the client in data migration work were introduced. The tool used to perform data migration in this thesis was JetBrains DataGrip. The introduction guide required imaginary startup data, which described the system from where data migration was performed. It was decided to make this startup data with Mockaroo.com, and the basic instructions for using this site were also included.</p> <p>The implementation phase started with designing the structure of introduction guide and designing the generic startup data with the selected service. The idea was to make the startup data an easy to store solution for use in the technical part of the introduction guide. In this thesis data migration was implemented between the startup data and the database structure of the client's ERP system, using the tools and practices used by the client. This data migration was recorded as a step-by-step introduction guide to support existing practices and materials of the client.</p> <p>As a result, the technical introduction guide for the client was recorded to support the orientation of the client's employees to data migration. At the same time the author of this thesis introduced himself to this job. The introduction guide includes the theoretical part, the technical implementation and the process and practices of the client's data migration.</p>	
<p>Keywords Data migration</p>	

SISÄLTÖ

1	JOHDANTO	6
1.1	Tilaaja	6
1.2	Työn kuvaus ja tavoitteet	6
1.3	Tutkimusmenetelmä.....	6
1.4	Suunnittelu.....	7
1.5	Perehdytys	7
2	DATAMIGRAATIO JA KÄYTETYT TYÖKALUT	8
2.1	Datamigraatio.....	8
2.2	Datamigraation prosessit	8
2.2.1	Suunnittelu.....	9
2.2.2	Datan valmistelu.....	11
2.2.3	Migraation suunnittelu	11
2.2.4	Migraation toteutus.....	12
2.2.5	Testaus.....	12
2.2.6	Tarkastus ja huolto	12
2.3	DataGrip.....	12
3	TOTEUTUS.....	14
3.1	Alkudatan luonti / Mockaroo	14
3.2	Alkudatan luonti.....	14
3.2.1	Ongelmatilanne Mockaroo	17
3.3	DataGrip asennus / alkudatan sisäänajo	18
3.3.1	Alkudatan sisäänajo DataGrip	20
3.4	Data mapping	21
3.4.1	Data mapping toteutus	22
4	TULOKSET	26
5	POHDINTA.....	27
6	LÄHDELUETTELO.....	28

KUVALUETTELO

KUVA 1. Työnopastuksen tarve. (STTK, ei pvm)	7
KUVA 2. IBM:n ehdotus kolmivaiheisesta datamigraatio prosessista. (Warmuth A., 2015, ss. 17-20)	9
KUVA 3. Kate Prohorchik A data migration strategy 6 steps. (Prohorchik, 2021)	9
KUVA 4. Lähtöjärjestelmän tietokannan rakennekaavio.....	15
KUVA 5. Esimerkki HENKILOT-taulun luonnista Mockaroo-palvelussa.	15
KUVA 6. Datasetin luominen Mockaroo-palvelussa.....	16
KUVA 7. ASIAKAS-taulun luonti ja sen relaatio HENKILOT-tauluun Mockaroo-palvelussa.	16
KUVA 8. Datan lataaminen DATASETS-osiosta Mockaroo-palvelussa.....	17
KUVA 9. Mockaroo-sivuston vaihtunut datasetin-toiminnon tuetut tiedostomuodot.	19
KUVA 10. Asiakas-taulun SQL-tiedoston Insert-lauseet.	18
KUVA 11. DataGrip ensimmäinen käynnistys.	18
KUVA 12. DataGrip projektin luonti.....	19
KUVA 13. Tietokantayhteyden luonti ja DataGrip-sovelluksen tuetut tietokannat.	19
KUVA 14. MySQL-tietokantayhteys: General-välilehti.	20
KUVA 15. Aloitusdatan lauseiden sisäänajo	20
KUVA 16. DataGrip aloituslauseiden sisäänajon tulokset	21
KUVA 17. Aloitusdatan tietokannan aputaulujen luontilauseet.	23
KUVA 18. Export Data -ominaisuus, SQL-Insert-Multirow	23
KUVA 19. Esimerkki ASIAKAS- ja HENKILO-taulujen SELECT-lauseista	24
KUVA 20. Tietojen kartoittaminen	25
KUVA 21. Kohdejärjestelmän taulujen numeroinnin tarkistuslause konversiolauseita varten	25

1 JOHDANTO

1.1 Tilaaja

Opinnäytetyön toimeksiantajana toimii jyvaskyläläinen Pinja Digital Oy, joka on osa Pinja-konsernia. Pinja Digital tuottaa palveluita IT-konsultoinnin ja -palveluiden sekä sovellusten ja ohjelmistojen toimialoilla. Toimeksiantajalla oli tarve kehittää tekninen ohjeistus ja materiaali tukemaan data migraation perehdytysprosessia data migraation työssä.

Pinja, alun perin Protacon on perustettu vuonna 1990. Pinja muodostui, kun Protacon ja siihen vuosien varrella liitetyt SWD, ARROW, Netwell, Descal, Powen sekä Vision Systems yhdistyivät yhteisen Pinja-nimen alle. Pinja tarjoaa asiakkailleen SaaS- ja BI-ratkaisuja sekä tuottaa asiakkailleen pilvi-alusta-, ylläpito-, ICT-, Tietoturva- ja tukipalveluita tuotantoympäristöihin. Pinja-konserni on kansainvälinen organisaatio, joka palvelee asiakkaitaan maailmanlaajuisesti yli 30 maassa. Suomessa Pinja työllistää yli 550 henkilöä. (Pinja Oy, ei pvm)

1.2 Työn kuvaus ja tavoitteet

Toimeksiannon aihe pohjautuu toimeksiantajan tarpeisiin kehittää datamigraatio projekteihinsa tekninen materiaali ja ohjeistus, kuinka datamigraatioita toteutetaan toimeksiantajan toiminnanohjausjärjestelmään. Opinnäytetyön materiaali sisältää datamigraatio -perehdytyksen teknisen puolen ohjeistuksen, materiaalin ja testimigraation toteutuksen.

Tämän opinnäytetyön tavoitteena on luoda toimeksiantajalle selkeä tekninen osuus toimeksiantajan datamigraatiotyöhön tukemaan toimeksiantajan jo olemassa olevaa perehdytysmateriaalia. Työn osuus perehdytysmateriaalista sisältää teknisen osuuden. Työ sisältää tilaajan datamigraatiotyön yleisten käytäntöjen lisäksi opastusta DataGrip -ohjelmiston peruskäyttöön, joka on olennaista tämän työn toteutettavassa datamigraatiossa.

Työssä tehdään datamigraatio itse generoidun aloitusdatan ja kohde järjestelmän eli tilaajan toiminnanohjausjärjestelmän välillä. Työn vaiheet dokumentoidaan tilaajan käyttöön. Työssä hyödynnetään toimeksiantajan datamigraatiotyön käytäntöjä, materiaaleja, kuten tietokantarakennetta, ja periaatteita, jotka on tarkoitettu vain toimeksiantajan käyttöön. Perehdytys dokumentin lisäksi työn tarkoituksena on kouluttaa opinnäytetyöntekijä toimeksiantajan datamigraatiotiimiin.

1.3 Tutkimusmenetelmä

Tämä työ on rakenteeltaan toiminnallinen kehitystyö, koska tavoitteena oli kehittää toimeksiantajan datamigraatiotyön perehdytysmateriaaleja jalostamalla prosessia teknillisellä ohjeistuksella. Työllä pyrittiin ratkaisemaan toimeksiantajan ongelma ja tehdä uudelle työntekijälle selkeä materiaali, jolla työntekijä oppii toimeksiantajan peruskäytännöt datamigraatiotyössä. Työn tutkimuksessa hyödynnettiin aiheeseen liittyviä tieteellisiä artikkeleita, tutkimuksia ja kirjallisuutta.

Työ rajattiin vain teknilliseen osuuteen ja toteutukseen. Tämä opinnäytetyö ei käsittele työntekijän perehdyttämistä vaiheittain, eikä sisällä ohjeistusta onnistuneeseen perehtymiseen, eikä HR käytäntöjä uuden työntekijän perehdytykseen yritykseen. Aiherajaus on siis vain perehdytysprosessin teknillinen osuus, toteutus ja ohjeistus, joka pätee toimeksiantajan datamigraatiotyössä.

1.4 Suunnittelu

Työn suunnittelu aloitettiin yhdessä tilaajan Pinja Digital Oy:n kanssa. Suunnittelun aloituspalaverissa pohdittiin työn tarvetta ja työn rajaukset. Aloituspalaverin jälkeen suunnittelupalavereita pidettiin tilaajan kanssa, joissa määriteltiin tarkemmin teknilliset rajaukset, käytettävät työkalut, käytettävät proseduurit sekä olemassa olevien materiaalien ja salassa pidettävän aineiston käyttöoikeudet ja rajaukset.

1.5 Perehdytys

Perehdyttäminen sisältää kaksi pääkäsitettä, joihin perehdyttäminen jaottuu, perehdyttäminen ja työnopastus. Perehdyttäminen sisältää ”Talo tutuksi” menetelmän, jossa tehdään yritys, työyhteisö ja työympäristö tutuksi. Tämä sisältää kaikki toimenpiteet, joilla uusityöntekijä oppii tuntemaan yrityksen ja työpaikkansa. Työnopastus sisältää työtehtävien tarkemman perehdyttämisen ja kouluttamisen eli työnopastus liittyy itse työn tekemiseen. (Työturvallisuus-keskus, ei pvm)

Kuten kappaleessa 1.3 Tutkimusmenetelmä on todettu, rajataan työ vain teknilliseen osuuteen ja toteutukseen eli työ on rajattu perehdytyksessä työnopastukseen. Tämän työn tulos on suunnattu perehdytystilanteeseen, jossa työntekijän työkuva vaihtuu, mutta toimeksiantaja voi tarvittaessa hyödyntää materiaalia myös uuden työntekijän perehdytyksessä data migraatio työhön.

Työnopastusta tarvitaan, kun:

- työtehtävät vaihtuvat
- työmenetelmät muuttuvat
- hankitaan ja otetaan käyttöön uusia koneita, laitteita ja aineita
- työ toistuu harvoin
- turvallisuusohjeita laiminlyödään
- työpaikalla sattuu työtapaturma tai havaitaan ammattitauti
- annetussa työnopastuksessa havaitaan puutteita
- tilanne poikkeaa tavanomaisesta
- havaitaan virheitä toiminnassa ja puutteita tuotteiden ja palvelujen laadussa.

Lähde: Työturvallisuuskeskus

Kuva 1. Työnopastuksen tarve. (STTK, ei pvm)

2 DATAMIGRAATIO JA KÄYTETYT TYÖKALUT

2.1 Datamigraatio

Datamigraatio on prosessi, jossa siirretään dataa paikasta -, formaatista -, tai sovelluksesta toiseen. Datamigraatiossa tietojen siirto sisältää datan profiloinnin, puhdistuksen, validoinnin ja jatkuvan tiedon laadunvarmistusprosessin kohdejärjestelmässä, jotta voidaan todeta datamigraatio onnistuneeksi. Yleisesti voisi sanoa datamigraation olevan laajemman tiedonsiirtoskenaario kokonaisuuden ensimmäinen vaihe monimutkaisessa prosessissa.

ETL-prosessien (Extract = talteen otto, Transform = muunnos, Load = lataus) mukaisesti kaikkiin datamigraatioihin sisältyy vähintään muunnos- ja latausvaiheet, eli Transform ja Load. Transform-vaiheella tarkoitetaan datan muuntamista ja rakenteen muokkaamista kohdejärjestelmään sopivaksi. Load-vaiheella taas tarkoitetaan datan latausta kohde ympäristöön. Extract-vaihe on datan talteen ottamista eli ulos vientiä esimerkiksi nykyisestä datakeskuksesta.

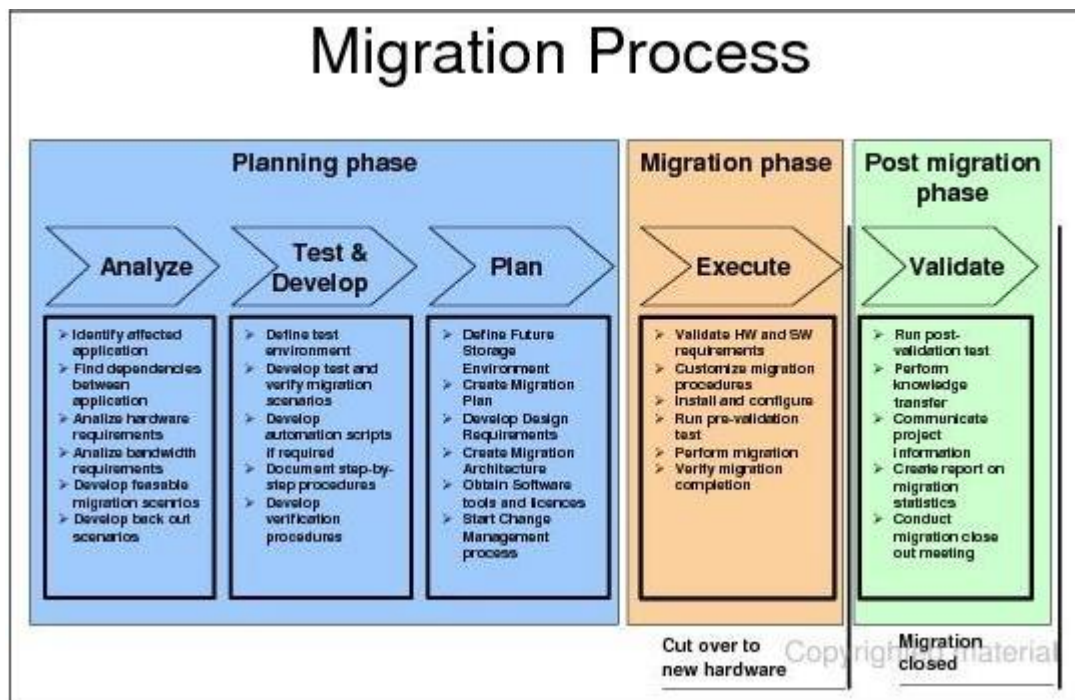
Datamigraatiot kategorisoidaan kolmeen tyyppiin:

1. Tallennustilan siirto. Datan siirron tarve keskittyy olemassa olevan datan taulujen laajentamista nykyaikaisempaan taulurakenteeseen, jotta useampi järjestelmä pystyy käyttämään dataa. Tämä tarjoaa nopeamman ja tehokkaamman suorituskyvyn sekä kustannustehokkaamman skaalauksen.
2. Pilvipalveluun siirto. Prosessi, jossa olemassa oleva data, kuten paikallisen datakeskuksen data ja sen rakenne, siirretään pilvipalveluun. Prosessi voi olla myös pilvipalveluiden välinen datan ja tallennustilan siirto.
3. Sovellussiirto. Sovellussiirto-migraatiossa sovellusohjelma siirretään ympäristöstä toiseen. Voi sisältää koko sovelluksen siirtämisen paikallisesta datakeskuksesta pilveen, pilvien välillä siirtämistä tai sovelluksen datan siirtämistä uuteen sovellukseen.

Nykyään datamigraatio on usein aineiston eli datan siirtämistä tietojärjestelmästä toiseen. Syynä voi olla esimerkiksi vanhan tietojärjestelmän uusiminen tehokkaampaan järjestelmään tai paikallisen datan siirtäminen pilvipalveluun. (Talend, ei pvm), (NetApp, ei pvm)

2.2 Datamigraation prosessit

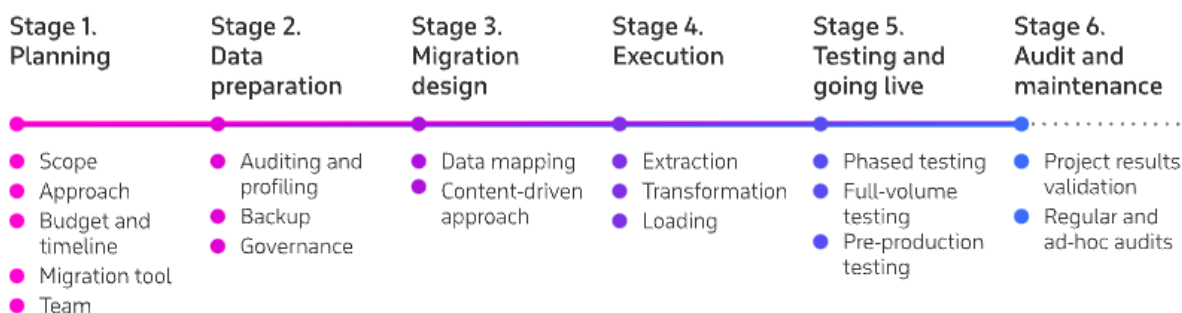
Teknologia yritys IBM:n mukaan useimmat datamigraatiotyöt koostuvat kolmesta vaiheesta: suunnittelu-, migraatio- ja migraation jälkeisestä vaiheesta. Nämä kolme vaihetta taas pilkotaan pienempiin kokonaisuuksiin. Näitä pienempiä vaiheita IBM:n datamigraatioprosessissa on yhteensä viisi. Suunnitteluvaihe sisältää analysoinnin, testauksen ja tuotannon, sekä datamigraation suunnittelun. Migraatiovaihe sisältää varsinaisen datamigraation toteutuksen ja migraation jälkeisessä vaiheessa taas validoidaan data varmistamalla sen migraation valmistuminen kohdejärjestelmään sekä kaikki kohdejärjestelmän toiminnallisuudet. (Warmuth A., 2015, ss. 17-30)



Kuva 2. IBM:n ehdotus kolmivaiheisesta datamigraatio prosessista. (Warmuth A., 2015, ss. 17-20)

Kate Prohorchik käy blogikirjoituksessaan "A data migration strategy in six steps" läpi datamigraatio strategian kuudessa vaiheessa 1. Suunnittelu, 2. Datan valmistelu 3. Migraation suunnittelu, 4. toteutus. 5. Migraation testaus ja 6. Tarkastus ja huolto. Nämä vaiheet sisältävät ja vastaavat teknologia yritys IBM:n ehdottamaa migraatio prosessin rakennetta. Käsittelen nämä Kate Prohorchikin vaiheet tarkemmin seuraavissa kappaleissa.

A data migration strategy



Kuva 3. Kate Prohorchik A data migration strategy 6 steps. (Prohorchik, 2021)

2.2.1 Suunnittelu

Suunnitteluvaihe kuuluu datamigraatiotyön kriittisimpiin ja tärkeimpiin vaiheisiin. Suunnitteluvaiheen tarkoitus on luoda selkeä suunta koko projektin ajaksi, se tarjoaa reitin ja määrittelee onnistumisen mittarit. Suunnitteluvaiheessa arvioidaan datamigraatio projektin laajuus, valitaan tapa lähestyä migraatiota, määritellään budjetti ja aika-arvio sekä valitaan käytettävät työkalut ja projektitiimi.

Ensiksi tulee suorittaa perusteellinen vertailu lähde- ja kohdejärjestelmien välillä. Yhdessä tilaajan ja käyttäjien kanssa tulee määritellä datan vähimmäistietomäärä kohdejärjestelmän käynnistämiseen

sopivaksi. Datamigraatiossa usein dataa joudutaan mukauttamaan vanhan ja uuden järjestelmän välillä niin, että tämä on tilaajalle ja kohdejärjestelmälle sopivan näköistä ja sisällöltään validia.

Datamigraation lähestymistavoiksi Prohorchik määrittelee kaksi ”big bang”- ja ”tricle one” -lähestymistavat. ”Big bang” tavalla tarkoitetaan lähestymistapaa, jossa datamigraatio suoritetaan yhtenä kokonaisuutena rajoitetun aikaikkunan sisällä. Tämä lähestymistapa voi sopia esimerkiksi pienille yrityksille, joiden data on suppeampaa ja seisokki aika minimaalista.

”Tricle one” -lähestymistavalla tarkoitetaan tapaa, jossa siirrytään järjestelmästä toiseen vaiheittain eli iteroimalla. ”Tricle one” -lähestymistavassa lähde- ja kohdejärjestelmät toimivat rinnakkain ja datan migraatio suoritetaan pienemmissä lohkoissa iteroimalla eli data pilkotaan pienempiin kokonaisuuksiin ja datamigraation työvaiheet toistetaan, kunnes kaikki haluttu data on siirretty kohdejärjestelmään.

Suurin osa yrityksistä käyttää migraatioissaan iteroimista, ”Tricle one” -lähestymistapaa, huolimatta siitä, että tämä lähestymistapa on monimutkaisempi prosessi, joka vaatii tarkkaa huolellista suunnittelua. Suunnittelun lisäksi iterointi vaatii huomattavasti enemmän vaivaa datan käyttäjiltä kuin myös migraatiotyön suunnittelijoilta, jotka joutuvat seuraamaan mitä dataa siirretään. Syy ”Tricle one” -lähestymistavan suosioon on se, ettei tämä lähestymistapa aiheuta vakavia seisokkiaikoja eikä toimintahäiriöitä, jos migraatio prosessi tehdään oikein. Lisäksi tämä lähestymistapa helpottaa virheiden ja ongelmatilanteiden havaitsemisen ja helpottaa näiden tilanteiden ratkaisemista prosessin varhaisissa vaiheissa.

Aikajana- ja budjettiarviot auttavat ymmärtämään projektin toteuttamiskelpoisuuden. Keskeisintä on ymmärtää siirrettävän datan määrä ja verkon nopeus, jotta voidaan arvioida datan siirtämiseen tarvittava aika. Tämän lisäksi on tärkeää tietää aiheuttaako datan siirto seisokkiaikaa ja suunnitella datan siirron ajankohta tämän mukaisesti. On yleistä, että datamigraation datan siirto toteutetaan esimerkiksi työajan jälkeisille tunneilla tai viikonlopuille, jolloin järjestelmien käyttöä ei ole tai se on vähäistä.

Budjettia suunniteltaessa tulee ottaa huomioon kaikkiin vaiheisiin arvioitu käytettävä aika eli datan siirtoon -, testaukseen-, migraation jälkeisiin tarkastuksiin – ja ylläpitoon käytettävä aika. Lisäksi tulee huomioida tiedonsiirtoratkaisun kustannukset ja projektitiimin maksut kaikissa vaiheissa, vaadittu datan validoinnin määrä ja mahdolliset seisokit. Huomioitavaa on se, että mitä enemmän on siirrettävää dataa, sitä korkeammat ovat projektikustannukset ja mahdolliset virheet. Myös tästä syystä ”Big bang” -lähestymistapa ei ole järkevää suurissa datamigraatioissa.

Mikäli datamigraatio toteutetaan yhtiön omalla datalla vanhasta järjestelmästä uuteen järjestelmään, tulee suunnitteluvaiheessa myös suunnitella vanhan järjestelmän käytön lopettaminen. Mikäli datamigraatio ja datamigraatiotiimi on ulkoistettu, tämä prosessi ei ole olennainen vaan vanhan järjestelmän käytön lopettamisen vastuu on datamigraation tilaajalla. Vanhan järjestelmän käytön päättämisessä on olennaista huomioida seuraavat:

- Lakkautta kaikki olemassa olevat aktiiviset sovelluslisenssit
- Hävitä tarpeettomat laitteet
- Peruuta vanhan järjestelmän tuki ja ylläpito

- Suunnittele henkilöstön uudelleen koulutus (Prohorchik, 2021)

2.2.2 Datan valmistelu

Kun datamigraation laajuus on hahmoteltu ja projektin toteutus suunniteltu voidaan aloittaa datan valmistelu vaihe. Datan valmistelu koostuu datan tarkastamisesta / validoimisesta, varmuuskopioimisesta ja hallinnoimisesta.

Vanha data tulee tarkastaa ja validoida. Suositeltavaa on käydä data tietoa alueittain ja tutkia yksityiskohtaisesti mahdollisten epä johdon mukaisuuksien, duplikaattien, virheellisen ja ristiriitaisen data varalta. Kaikki ongelmat tulee ratkaista ennen datan siirtoa uuteen järjestelmään. Vaiheen tarkoitus on puhdistaa virheellinen data ja tuoda kohdejärjestelmään validoitua oikeata dataa. Tämä vaihe voi olla erittäin työläs, joten yleensä datan tarkastukseen käytetään automaatiotyökalua.

Kun datan laatu on tarkastettu, on suositeltavaa varmuuskopioida data. Kaikkia ongelmatilanteita on mahdotonta ennakoita ja nämä tilanteet voivat johtaa tiedon menetykseen tai vioittumiseen. Tämän takia varmuuskopiointi toimii ylimääräisenä suojakerroksena, jos datan siirrossa tapahtuu virheitä, voidaan data palauttaa tarvittaessa.

Datan valmistelun hallinnoinnin osiossa tulee selvittää vanhasta järjestelmästä käyttäjien käyttöoikeudet datalle. Eli kenellä on oikeus käyttää, muokata ja poistaa mitään dataa. Nämä oikeudet tulee siirtää uuteen käyttöjärjestelmään, jotta datan käyttö ja oikeudet pysyvät samana myös uudessa järjestelmässä. Tämän lisäksi käyttäjät tietävät roolinsa ja vastuunsa tiedon siirron aikana ja sen jälkeen, tämä taas auttaa välttämään virhetilanteita ja tietoturva-aukkoja. (Prohorchik, 2021)

2.2.3 Migraation suunnittelu

Ennen varsinaista migraation toteutusta eli ETL-prosessia, tulee asiakkaan / tilaajan / datan käyttäjän kanssa pitää niin sanottu aloituspalaveri, jossa kartoitetaan datan säännöt "mapping rules". Palaverissa määritellään lähdejärjestelmän kentät vastaamaan kohdejärjestelmän kenttiä, jotta data on haluttua uudessa järjestelmässä.

Datan kartoittamisessa tulee pohtia metadataa ja datan sijainnin sisältöä ja pohtia sopiva lähestymistapa datan siirtoon näiden välillä. Metadata kuvaa usein lähtödatan sijaintia tai sijaintityypin ominaisuutta eli datatyyppiä. Sijainteja on esimerkiksi tiedosto, taulun nimi, taulun sarake ja datatyyppiä ovat sijainnissa olevan datan ominaisuus kuten integer (numero), date (päivämäärä) tai character (merkki). Sisältö tallennetaan näihin sijainteihin / kenttiin ja sisältää käyttäjän syötettä. Esimerkiksi taulu voi sisältää sarakkeen nimeltä email, jonka datatyyppi on character, tässä sarakkeessa on sisältö, joka vastaa käyttäjän syötettä eli hänen sähköpostiosoitettansa.

Ongelmia metadata lähtöisessä lähestymistavassa on oletukset taulurakenteen sisällöstä. Esimerkiksi aiemmin mainittu sähköposti tieto voi olla useassa eri sarakkeessa, kuten työsähköposti ja kotisähköposti. Näin ollen uuteen järjestelmään siirrettäessä voi data olla virheellistä, jos tämän sijainnin sähköposti ei ollut haluttu sisältö uuden järjestelmän vastaavaan sarakkeeseen. Kartoittamisvirheiden välttämiseksi on tärkeää suorittaa sisältöanalyysi tietojen tarkastuksen ja profiloinnin vaiheessa dataa valmistellessa, opinnäytetyön kappaleessa 3.2.2 Datan valmistelu. (Prohorchik, 2021).

2.2.4 Migraation toteutus

Kun työn laajuus, vaatimukset, tekijät, työkalut ja datan kartoitus on tehty, voidaan siirtyä datamigraation toteutukseen ETL-prosessina. ETL-prosessi sisältää datan lataamisen lähdejärjestelmästä (Extract), sen muokkaamisen (Transform) ja lataamisen kohdejärjestelmään (Load).

Toteutuksessa hyödynnetään aiempien vaiheiden määrittämiä kuten datamigraatioissa käytettävää migraatiotyökalua, jolla migraatiotekniset osat, kuten datan käsittely ja konversiot sekä datan sisäänajo tullaan toteuttamaan ja datamigraation suunniteltua datan kartoitusta. Kuten aiemmin kapaleessa 3.2.1 on todettu, suurin osa yrityksistä käyttää ”Trickle one” -lähestymistapaa datamigraatioissa eli iterointia. Data pilkotaan ja muokataan kohdejärjestelmään sopivaksi pienemmissä osissa ja ladataan iteroimalla nämä osat kohdejärjestelmään. (Prohorchik, 2021)

2.2.5 Testaus

Testauksen Prohorchik suosittelee aloitettavaksi heti kuin datan käsittely aloitetaan ja sitä tulisi jatkaa siitä datamigraatiotyön prosessin loppuun asti eli Prohorchik suosittelee testauksen aloittamisesta datan valmistelusta alkaen. Lisäksi jos datamigraatio toteutetaan vaiheittain, tulee jokainen osio testata erikseen, jotta virhetilanteet löydetään ajoissa ja datan laatu varmistuu. Mikäli virheitä löytyy, tulee data testata uudelleen korjauksen jälkeen. Kun datan siirto on valmis, tulee siirretyt tiedot varmistaa yksikkö ja järjestelmä täydellä kapasiteetti- ja osiotestillä. Huomioitavaa on, että täyden kapasiteetin testaaminen on myös tärkeää, sillä tällä tapaa voidaan löytää vielä virhetilanteita, joita ei osiotesteissä ole huomattu. Datamigraation testaukset toteutetaan testausympäristöön ja vasta kun virheitä ei löydy ja kaikki osapuolet ovat tyytyväisiä tuloksiin voidaan datan siirto aloittaa tuotantoympäristöön. (Prohorchik, 2021)

2.2.6 Tarkastus ja huolto

Tuotantoympäristön julkaisun jälkeen on tärkeää validoida datamigraatioprojektin tulokset ja seurata uuden järjestelmän suorituskykyä pidemmällä aikavälillä. Datamigraatio projektin päätteeksi on tärkeää suorittaa kattava tarkastus. Datan täytyy olla laadukasta ja tietoja ei saa puuttua. Lisäksi Prohorchik suosittelee säännöllisiä ”Ad hoc” -tarkastuksia varmistuakseen, että järjestelmä kattaa tiedon laajuuden ja data on yhä laadukasta ja oikeanlaista. ”Ad hoc” -tarkastusten tarkoitus datamigraation toteutuksen jälkeen on varmistaa, että kohdejärjestelmä pystyy käsittelemään datan onnistuneesti. ”Ad hoc” -tarkastuksilla voidaan esimerkiksi tarkistaa usein käytettyjen ja tärkeiden taulujen datan varmistuakseen, että lähde- ja kohdetiedot täsmäävät. (Prohorchik, 2021) (Katkamwar, 2016)

Ad hoc on latinaa ja tarkoittaa ”tätä tarkoitusta varten” ja sillä tarkoitetaan, että tietty tapahtuma on väliaikainen ja tarkoitettu tiettyyn tarkoitukseen. Eli tässä aiheytymässä tarkastukset tehdään todentamaan datan oikeellisuus varsinkin kriittisissä usein käytetyissä tauluissa. (Tietosanakirja-titanica, 2022)

2.3 DataGrip

DataGrip on JetBrainsin julkaisema maksullinen tietokanta-IDE (Integrated Development Environment, suom. Ohjelmointiympäristö) eli tietokanta työkalu, joka on suunnattu SQL-kehittäjien erityistarpeisiin. DataGrip ei ole pakollinen migraatiotyön toteutukseen vaan migraatiotyön voi toteuttaa

jollain toisella migraatiotyökalulla. Tilaaja on valinnut DataGripin käytettäväksi datamigraatiotyös-
sään, joten tämä opinnäytetyö ja siihen kuuluva datamigraatio toteutettiin DataGrip-työkalulla.
(JetBrains, ei pvm)

3 TOTEUTUS

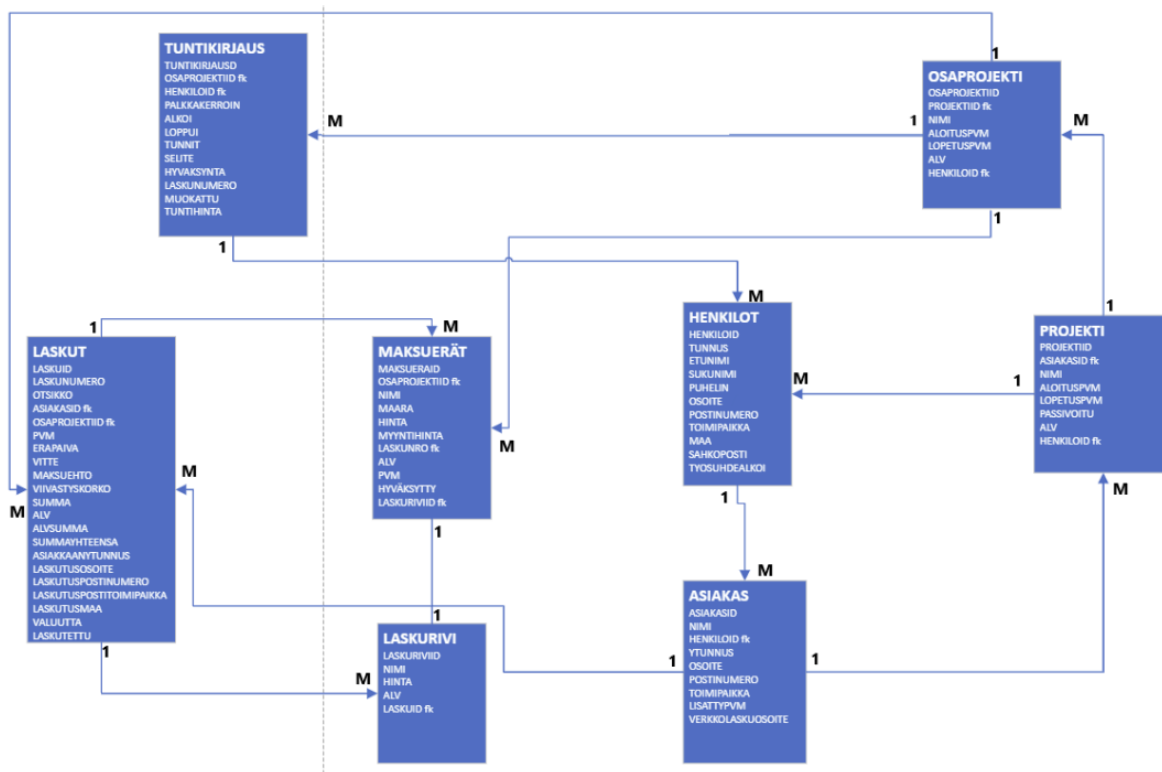
3.1 Alkudatan luonti / Mockaroo

Alkudatan luomiseen päädyttiin käyttämään Mockaroo-sivustoa. Aloitusdata luotiin generiseksi datarakenteeltaan kuvaamaan toiminnanhallintajärjestelmää, jotta tätä dataa pystyttiin esittämään opinnäytetyössä. Kohdejärjestelmän tietokantarakenne on salassa pidettävää tietoa, joten sitä ei esitetty opinnäytetyössä. Opinnäytetyöstä muodostetussa teknillisen toteutuksen dokumentissa on kokonaisvaltainen datamigraatio toteutus aloitusdatan ja kohdejärjestelmän välillä tilaajan käytettäväksi datamigraatiotyön perehdytykseen.

Mockaroo.com on sivusto, jossa voidaan luoda monenlaista testidataa käytettäväksi ja muokattavaksi. Data muodostetaan yksinkertaisesti valitsemalla datalle sarakkeiden nimet ja saraketta vastaavan tietotyyppin sivuston valmiiksi tehdyistä lukuisista valinnoista. Tyyppi-valinta generoi valmiiksi tiedolle datatyyppin (SQL-tallennuksessa) ja generisiä eri datarivejä, kuten valitessasi tyyppiä etunimen, generoi tämä useammasta valmiiksi syötetystä etunimestä riveille dataa. Lisäksi sarakkeelle voidaan määritellä haluamiasi ehtoja ja sivustolla voidaan määritellä rivien määrä, tallennettava tiedostomuoto ja tiedostomuodolle tyypillisiä ominaisuuksia, kuten SQL-tiedostolle taulun nimi ja valinta lisätäänkö tallennettavaan tiedostoon SQL-tilauksen luontilauseet. Kun sarakkeet on määritelty ja haluttu formaatti, ehdot ja rivien määrä on valittu, voidaan dataa esikatsella "Preview"-painikkeella, ladata "Download data" -painikkeella ja tallentaa malliksi "Save this schema" -painikkeella. "More"-painikkeesta löytyy valinnat datan tuontiin. Data voidaan tuoda palveluun JSON-, Excel/CSV-, sql create table – muodossa ja suora toistaa MQTT verkkoprotokollalla.

3.2 Alkudatan luonti

Aloitusdata koostuu kahdeksasta taulusta, joiden väleillä on relaatioita: ASIAKAS, PROJEKTI, OSA-PROJEKTI, HENKILOT, TUNTIKIRJAUS, LASKUT, LASKURIVIT ja MAKSUERAT. Helpottaakseen aloitusdatan luontia, luotiin Mockaroo-palveluun ilmainen käyttäjätunnus, jotta aloitusdatan tauluja, dataa ja taulujen kaavioita pystyttiin hallitsemaan paremmin tallentamalla nämä käyttäjätunnukselle. Käyttäjätunnus ei ole pakollinen Mockaroo-palvelussa datan luomiselle. Alkudataan luotiin 500kpl asiakasta, 1000kpl käyttäjää (Henkilot-taulu), 800kpl projektia, 1000kpl osaprojektia, 500kpl laskua, 1000kpl laskuriviä, 500kpl maksuerää ja 1000kpl tuntikirjausta.



Kuva 4. Lähtöjärjestelmän tietokannan rakennekaavio.

Asiakas-taulu sisältää kuvitteellisen lähtöjärjestelmän asiakastiedot. Asiakas-taulun asiakasvastaava-sarakkeelle luotiin relaatio Henkilot-taulun id-sarakkeeseen. Ennen Asiakas-taulun muodostamista täytyi muodostaa Henkilot-taulu, jotta relaatioavain oli käytettävissä Asiakas-taulua muodostaessa.

Mockaroo-palvelussa voidaan tallentaa taulujen kaaviot myöhemmin käytettäväksi "Save this schema" -painikkeella. Tätä ominaisuutta tarvitaan myös, kun tehdään tauluille relaatio Mockaroo-palvelussa.

Field Name	Type	Options
ID	Row Number	blank: 0% Σ ×
TUNNUS	Character Sequence	**@@" blank: 0% Σ ×
ETUNIMI	First Name	blank: 0% Σ ×
SUKUNIMI	Last Name	blank: 0% Σ ×
PUHELIN	Phone	format: ###-###-#### blank: 0% Σ ×
OSOITE	Street Address	blank: 0% Σ ×
POSTINUMERO	Postal Code	blank: 0% Σ ×
TOIMIPAikka	City	blank: 0% Σ ×
MAA	Custom List	Suomi random blank: 0% Σ ×
SAHKOPOSTI	Email Address	blank: 0% Σ ×

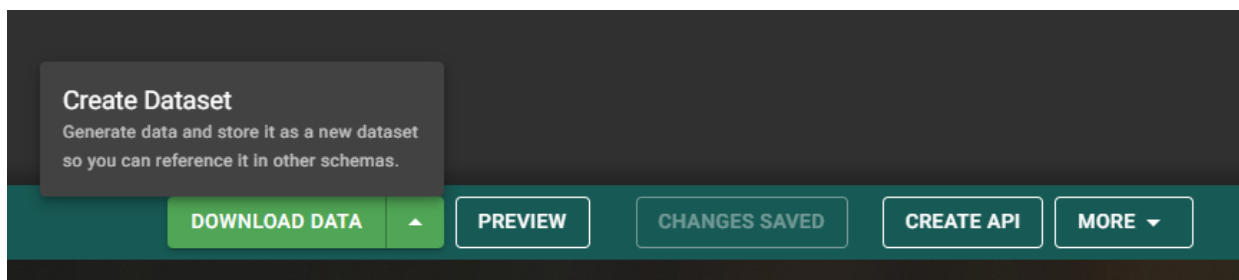
Rows: 1000 Format: SQL Table Name: HENKILOT include CREATE TABLE

Append Dataset: choose a dataset...

DOWNLOAD DATA PREVIEW SAVE THIS SCHEMA MORE

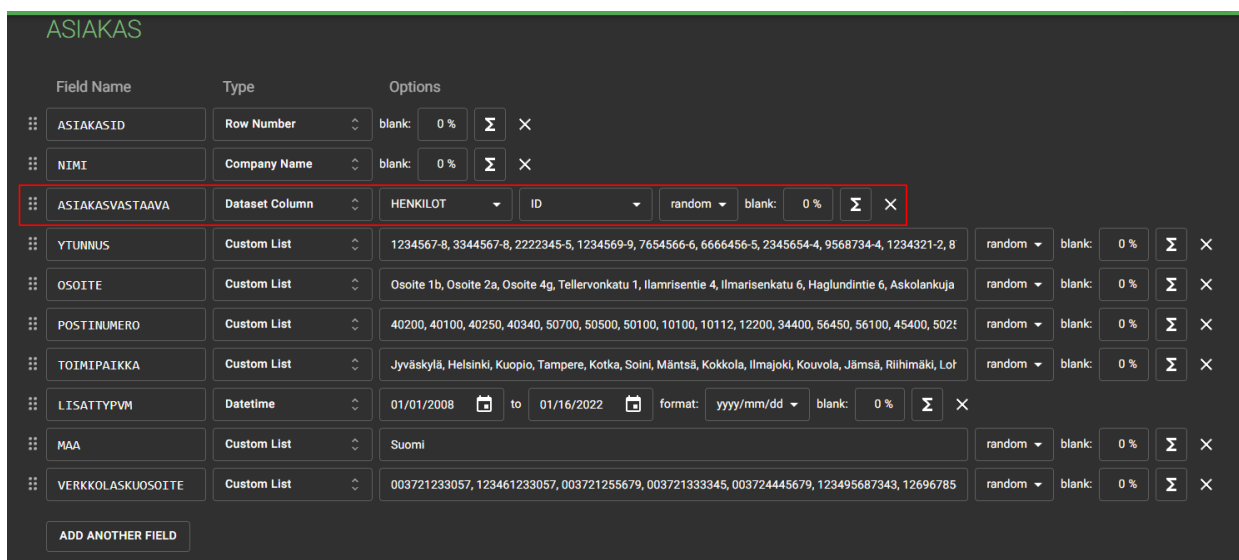
Kuva 5. Esimerkki HENKILOT-taulun luonnista Mockaroo-palvelussa.

Kun Henkilot-taulun kaavio oli tallennettu, voitiin data luoda Mockaroo-palveluun datasettinä "Download data" -painikkeen valintaruudusta valitsemalla "Create Dataset".



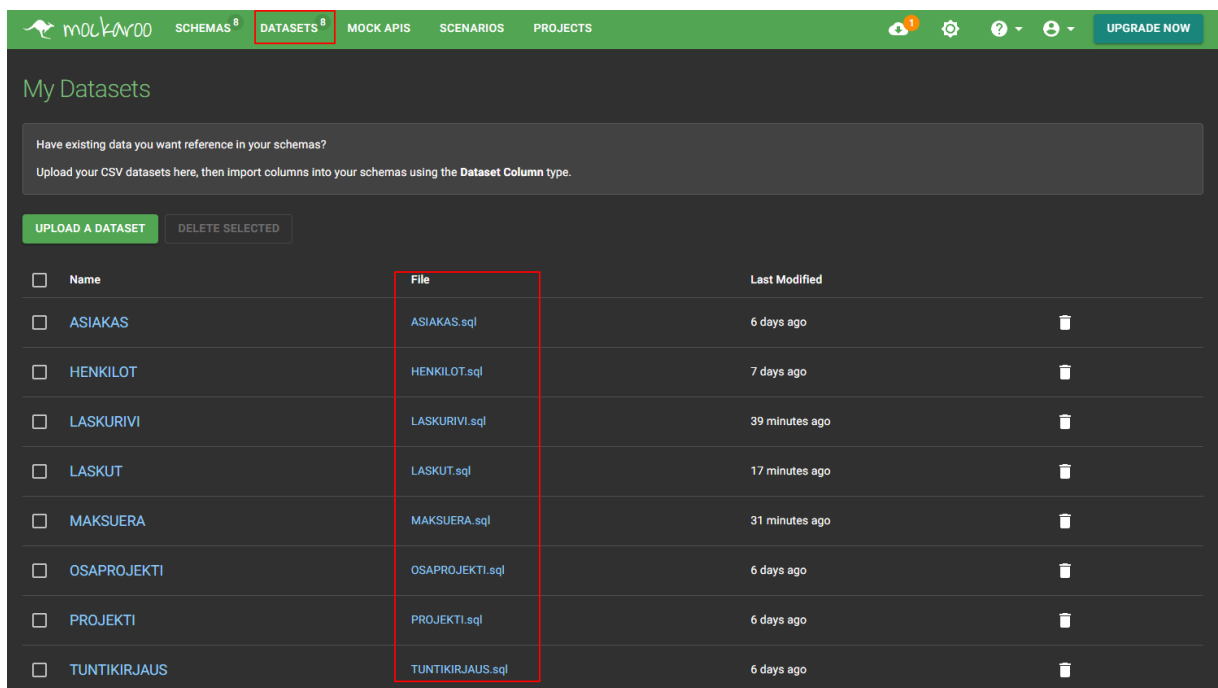
Kuva 6. Datasetin luominen Mockaroo-palvelussa.

Asiakas-taulun asiakasvastaava-sarakkeelle voitiin nyt muodostaa relaatio Henkilot-taulun id-sarakkeelle. Valitsemalla "Type"-valinnaksi "Dataset Column", jonka jälkeen "Options"-valintaan voitiin valita haluttu taulu ja taulun sarake. Sama prosessi toteutettiin kaikille tauluille.



Kuva 7. ASIAKAS-taulun luonti ja sen relaatio HENKILOT-tauluun Mockaroo-palvelussa.

Kun kaikki taulut ja datasetit oli luotu Mockaroo.com sivustolla, voitiin datasetit tallentaa sql tiedosto muodossa sivuston DATASETS-osiossa klikkaamalla File-sarakkeen linkistä. Vaihtoehtoisesti data olisi voitu ladata suoraan jo taulun sarakkeiden muodostamisen jälkeen "DOWNLAOD DATA"-valinnalla.

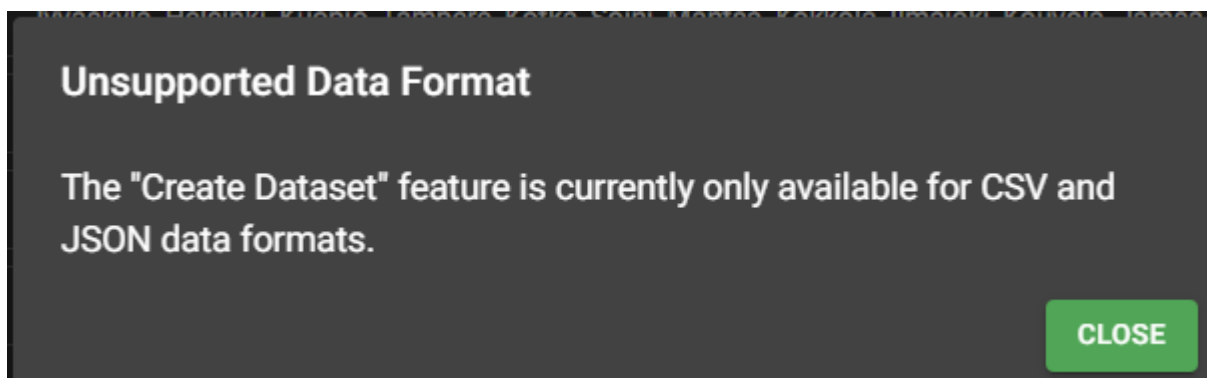


Kuva 8. Datan lataaminen DATASETS-osiosta Mockaroo-palvelussa.

3.2.1 Ongelmatilanne Mockaroo

Mockaroo-sivusto on kätevä datan luomiseen suunniteltu sivusto, eikä datan muodostamiseen tarvita sovellusta. Ongelmaksi kuitenkin koitui relaatiosuhteiden luominen. Relaatiosuhde datasettiin ei onnistunut skeemasta, mikäli skeeman sarake ei vastaa datasetin skeeman sarakkeen nimeämistä. Eli esimerkiksi opinnäytetyön ASIAKAS-skeemasta datasettiä luodessa ei skeeman ASIAKASVASTAAVA-sarakkeeseen generoitunut haluttua dataa, koska sarake ei vastannut HENKILOT-skeeman ID-saraketta nimellisesti, johon tämä relatio oli asetettu. Ratkaisuna tähän täytyi vaihtaa ASIAKAS- ja HENKILOT-skeeman sarakkeet saman nimisiksi. Koska ASIAKAS-skeemassa oli jo ID-sarake, vaihdettiin ASIAKAS-skeeman ASIAKASVASTAAVA-sarake ja HENKILOT-skeeman ID-sarake HENKILOID nimiseksi. Sama toistettiin muidenkin relaatioiden osalta.

Toiseksi ongelmaksi koitui Mockaroo-sivuston uusi käytäntö, ettei SQL-tiedostomuodosta voinut enää toteuttaa datasettiä, joka oli mahdollista ennen näiden relaatioiden nimeämisen muutoksia. Nyt generoitui kuvan 9 mukainen virheilmoitus.



Kuva 9. Mockaroo-sivuston vaihtunut datasetin-toiminnon tuetut tiedostomuodot.

Tämän uuden dataset-toiminnon tuettujen tiedostomuotojen rajauksen myötä täytyi työssä keksiä uusi tapa muodostaa alitusdata. Koska Mockaroo-palvelussa oli tehty jo valmiiksi skeemat ja

Mockaroo-palvelussa datasettejä voidaan luoda muissa tiedostomuodoissa, kuten CSV, päädyttiin datasetit muodostamaan CSV-tiedostomuodossa. Kun datasetit oli muodostettu CSV-muodossa, voitiin CSV-tiedostomuotojen dataseteistä generoida relaatiot SQL-tiedostoihin ja ladata nämä suoraan "DOWNLOAD DATA" -toiminnolla. Kuvassa 10 kuvattu korjatut SQL-tiedostot, kun lauseet oli generoitu CSV-datasetistä.

```

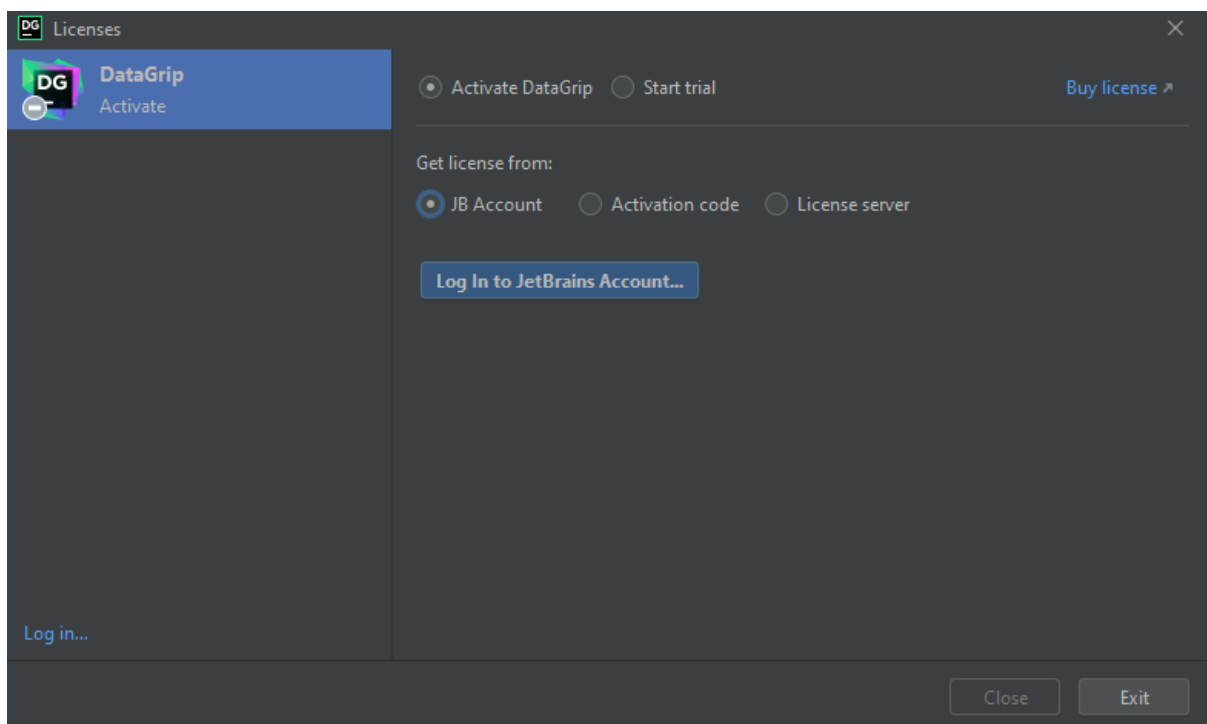
MAA VARCHAR(5),
VERKKOLASKUOSOITE VARCHAR(12)
);
insert into ASIAKAS (ASIAKASID, NIMI, HENKILOID, YTUNNUS, OSOITE, POSTINUMERO, TOIMIPAikka, LISATTYPVM, MAA, VERKKOLASKUOSOITE) values (1, 'InnoZ', '968', '1
insert into ASIAKAS (ASIAKASID, NIMI, HENKILOID, YTUNNUS, OSOITE, POSTINUMERO, TOIMIPAikka, LISATTYPVM, MAA, VERKKOLASKUOSOITE) values (2, 'Yakitri', '872',
insert into ASIAKAS (ASIAKASID, NIMI, HENKILOID, YTUNNUS, OSOITE, POSTINUMERO, TOIMIPAikka, LISATTYPVM, MAA, VERKKOLASKUOSOITE) values (3, 'Skyble', '619',

```

Kuva 10. Asiakas-taulun SQL-tiedoston Insert-lauseet

3.3 DataGrip asennus / alkudatan sisäänaajo

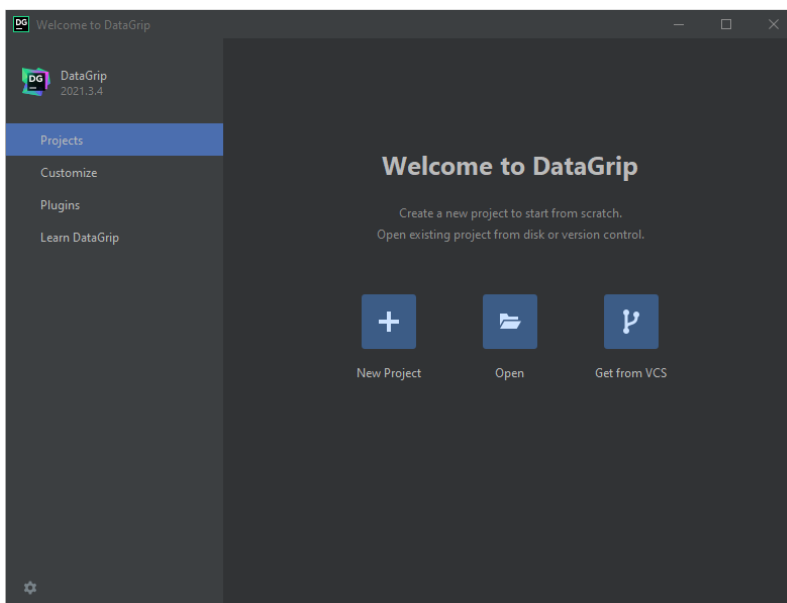
Kuten aikaisemmin mainitti, tilaaja käyttää migraatio työssään JetBrainsin julkaisemaa DataGrip-sovellusta sovelluksen lukuisten ominaisuuksien, kuten nopeiden tietokantayhteyksien takia. JetBrains tarjoaa myös ilmaista 30 päivän kokeilujaksoa DataGrip-sovelluksesta, joka voidaan asentaa osoitteesta <https://www.jetbrains.com/datagrip/>. DataGripin asennus on suoraviivainen prosessi, joka sisältää asennusavustajan, joten asennusta ei käydy tarkemmin tässä opinnäytetyössä. Asennuksen ja loppukäyttäjän lisenssisopimuksen jälkeen aukeaa DataGripin aktivointi ikkuna, jossa voidaan kirjautua jo olemassa olevalle JetBrains-käyttäjätunnukselle, syöttää ohjelmiston aktivointikoodi tai aloittaa ilmaiskokeilu, valinnat kuvassa 11.



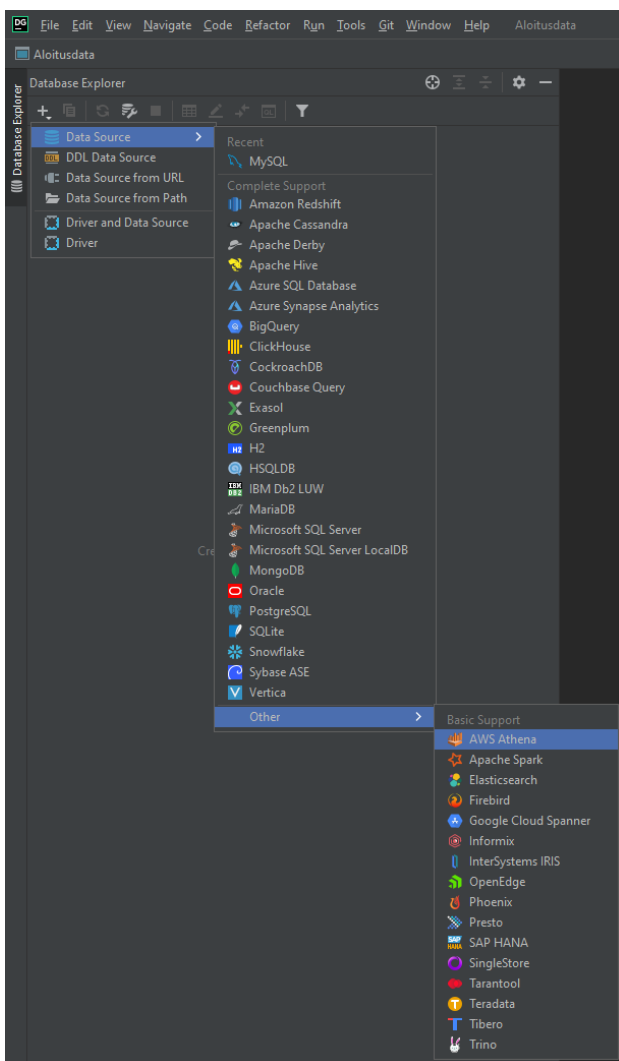
Kuva 11. DataGrip ensimmäinen käynnistys.

Kun lisenssi on syötetty tai ilmaiskokeilu aktivoitu, voidaan aloittaa DataGrip-sovelluksen käyttö. DataGrip käyttää projektirakennetta, joten ensiksi sovellukselle tulee määrittää projektin nimi tai valita olemassa oleva projekti (kuvio 12). Tämän jälkeen tulee määrittellä tietokanta yhteys. DataGrip tukee täydellisesti 26 eri tietokantarakennetta ja olennaisten ominaisuuksien osalta 16 muuta tietokantaa. Tuetut tietokannat ja yhteyden luomiseen pääsee klikkaamalla +-valikkoa ja valitsemalla "Data

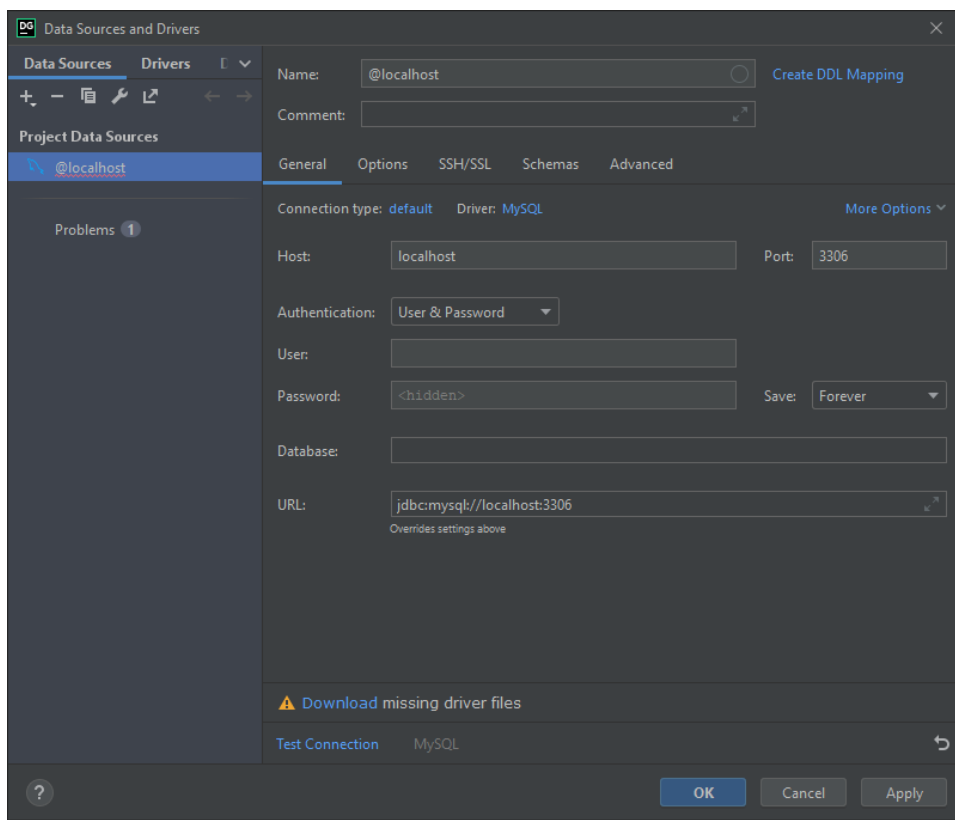
source” -valikko. ”Data source” -valikossa ”Other”-osiossa on perusominaisuuksien osalta tuetut 16 tietokantaa (kuvio 13). Tietokantayhteyden esittäminen tilaajan tietokantaan on salassa pidettävää tietoa, joten kuviossa 14 esitetään yleisen MySQL-tietokannan yhteyden luomisen General-välilehti ja sen valinnat.



Kuva 12. DataGrip projektin luonti.



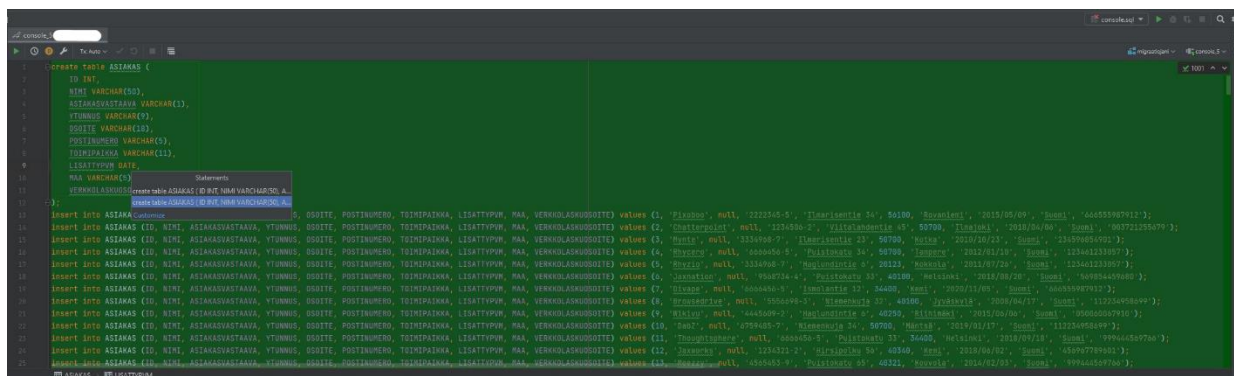
Kuva 13. Tietokantayhteyden luonti ja DataGrip-sovelluksen tuetut tietokannat



Kuva 14. MySQL-tietokantayhteys: General-välilehti.

3.3.1 Alkudatan sisäänajo DataGrip

Kun tietokanta yhteys oli luotu, ajettiin Mockaroo-palvelussa muodostetut SQL-tiedostot tietokantaan. DataGrip mahdollistaa useamman SQL-lauseen ajamisen kerrallaan tietokantaan, tämä onnistuu valitsemalla ensimmäinen ajettava lause klikkaamalla hiiren vasemmalla painikkeella lauseesta. Painalluksen jälkeen lauseen ympärille muodostuu vihreä neliö ilmoittamaan visuaalisesti, että kyseinen lause on valittu ajettavaksi. Pikanäppäimillä `ctrl + enter` voidaan avata lauseen ajoikkuna, josta voidaan valita merkitty lause tai kaikki kyseisen Query consolen lauseet ajettavaksi. Kun lause on ajettu onnistuneesti, merkitsee DataGrip IDE lauseen eteen vihreän oikein-merkin. Lisäksi DataGripissä on query consolen tulos -ikkuna, josta voi nähdä syötetyt rivit ja niiden toteutuneet tiedot.



Kuva 15. Aloitustietokannan lauseiden sisäänajo

```

6619 ✓ insert into TUNTIKIRJAUS (ID, OSAPROJEKTI, HENKILO, PALKKAKERROIN, ALKOI, LOPPUI, TUNNIT, SELITE, HYVAKSYNTA, LASK
6620

[2022-02-06 19:02:02] 1 row affected in 66 ms
migratiojoni> insert into TUNTIKIRJAUS (ID, OSAPROJEKTI, HENKILO, PALKKAKERROIN, ALKOI, LOPPUI, TUNNIT, SELITE, HYVAKSYNTA,
[2022-02-06 19:02:03] 1 row affected in 66 ms
migratiojoni> insert into TUNTIKIRJAUS (ID, OSAPROJEKTI, HENKILO, PALKKAKERROIN, ALKOI, LOPPUI, TUNNIT, SELITE, HYVAKSYNTA,
[2022-02-06 19:02:05] 1 row affected in 64 ms
migratiojoni> insert into TUNTIKIRJAUS (ID, OSAPROJEKTI, HENKILO, PALKKAKERROIN, ALKOI, LOPPUI, TUNNIT, SELITE, HYVAKSYNTA,
[2022-02-06 19:02:06] 1 row affected in 75 ms
migratiojoni> insert into TUNTIKIRJAUS (ID, OSAPROJEKTI, HENKILO, PALKKAKERROIN, ALKOI, LOPPUI, TUNNIT, SELITE, HYVAKSYNTA,
[2022-02-06 19:02:07] 1 row affected in 66 ms
migratiojoni> insert into TUNTIKIRJAUS (ID, OSAPROJEKTI, HENKILO, PALKKAKERROIN, ALKOI, LOPPUI, TUNNIT, SELITE, HYVAKSYNTA,
[2022-02-06 19:02:09] 1 row affected in 64 ms

```

Kuva 16. DataGrip aloituslauseiden sisäänajon tulokset

3.4 Data mapping

Tietojen siirtämisen ja yhdistämisen toteutuksessa tarvitaan etenemissuunnitelma, jotta tiedot saadaan siirrettyä oikein. Tiedonhallintaprosesseissa, joissa tietoa siirretään tai käsitellään järjestelmien tai tietojen sijaintien välillä, tarvitaan data mapping -käytäntöä eli tietojen kartoittamista. Tietojen kartoittamisen laatu korreloi datan oikeellisuuden kohdejärjestelmässä tai tietojen sijainnissa. Lähtödatan tulee olla yhdenmukaista kohdedatan kanssa, esimerkiksi lähtöjärjestelmä voi käyttää kaupungin lyhennettä, kun taas kohdejärjestelmässä sama tieto esitetään kaupungin koko nimen mukaan. Tietojen kartoitus on avainasemassa tiedon oikeellisuudessa myös datamigraatiotyön toteutuksessa. Tietojen kartoituksen vaiheet ovat

1. Määrittäminen:
Määritä siirrettävä datan taulut, taulujen sarakkeet ja sarakkeiden muoto siirron jälkeen.
2. Tietojen yhdistäminen:
Yhdistä lähtöjärjestelmän lähtökentät kohdejärjestelmän kenttiin sopiviksi.
3. Muunnos:
Jos sarake vaatii muunnoksen lähtöjärjestelmästä kohdejärjestelmään, tarvitaan se muuntaa vastaamaan kohdejärjestelmän saraketta. Sarake-muunnokset toteutetaan koodilla.
4. Testaus:
Suorita siirto testausjärjestelmään ennen varsinaista kohdejärjestelmää. Testausjärjestelmästä voidaan tarkistaa datan toimivuus, mikäli data muodostuu virheellisesti, korjataan kartoitus virheellisten tietojen osalta.
5. Käyttöönotto:
Kun tietojen muunnos on todettu suunnitellun mukaiseksi, ajoita siirto- tai integrointi tapahtuma kohteeseen.
6. Ylläpito ja päivitys:
Mikäli kyseessä on datan integrointi, joka vaatii jatkuvaa tietojen siirtoa ja – kartoitusta, on tietokartta taipuvainen muutoksille ja vaatii päivityksiä. Kun uusia tietolähteitä lisätään, ne tai kohteen vaatimukset muuttuvat, tulee tietokartta päivittää näiden muutosten mukaisesti.

Tietojen kartoittamista voidaan myös automatisoida ja käyttää tiedon kartoittamiseen työkaluja, jotka tehostavat tiedonkartoituksen prosessia. Tällaisia työkaluja ovat esimerkiksi kehittyneet pilvi-

pohjaiset tiedonkartoitustyökalut, jotka auttavat yritystä visualisoimaan tietojen kartoituksen ja toteuttaa tiedonkartoituksen pienellä budjetilla. Tiedonkartoitustyökalut ovat erinomaisia jatkuvissa dataintegraatioissa, joissa myös tietojen kartoituksen vaihe 6. ylläpito ja päivitys on olennainen. Tärkeää on valita omaan työhön sopiva kartoitustyökalu, joka tukee laajasti eri tiedostoformaatteja. (Talend, ei pvm)

3.4.1 Data mapping toteutus

Tämän työn tietojen kartoitus ja datan sisäänajo toteutettiin manuaalisesti SQL-syntaksilla DataGrip-sovelluksella. Toteutuksessa hyödynnettiin kahta aputaulua, joiden tarkoituksena oli juoksuttaa taulun rivien yksilöivälle tiedolle uudet kohdejärjestelmän yksilöivät tiedot. Aloitustiedot ajettiin omaan tietokantaan ja kohdejärjestelmäksi tehtiin tilaajan mallitietokanta. Aloitustiedot tehtiin kuvitteelliset asiakkaan toimittamat määritykset eli ehdot, joiden mukaan datamigraatio toteutetaan. Nämä ehdot lähtödataan olivat seuraavat:

- Passiiviset projektit ja osaprojektit tuodaan vain, jos lopetuspäivämäärä on 01.01.2018 jälkeen.
- Näiden osalta jätetään myös laskut, laskurivit, tuntikirjaukset ja maksuerät datamigraation ulkopuolelle.

Aputaulut ja niiden sarakkeet:

MIGRATION_ID_VALUES-taulu:

- ID: migration_id_values-taulun rivin yksilöivä ID
- TABLE_NAME: Kohdejärjestelmän taulun nimi
- MAXID_START: Kohdejärjestelmän ensimmäisen syötettävän rivin ID
- MAXID_END: Kohdejärjestelmän viimeisen syötettävän rivin ID
- UPDATED: Taulun luonti / päivitys aikaleima

MIGRATION_MAP-taulu:

- TAULU: relaatio migration_id_values-taulun käsiteltävän taulun ID-tietoon.
- ID: lähtödatan kyseisen (käsiteltävän) taulun rivin yksilöivä tieto.
- ID_NEW: kohde järjestelmän vastaavan käsiteltävän taulun rivin ID lähtödatan riville

MIGRATION_ID_VALUES-taulun tarkoituksena oli ylläpitää ID-tietoja ja luoda relaatio näiden tietojen välillä. Tauluun haettiin kohdejärjestelmän maksimi ID-tiedot ja kohdejärjestelmän päivitettävien taulujen nimitiedot. Lisäksi MAXID_END- ja UPDATED-sarakkeet toimivat varmistena tehdystä työstä ja näiden sarakkeiden tarkoitus oli helpottaa sisään ajettujen rivien määrän tarkistusta.

MIGRATION_MAP-taulu toimi tietojen kartoitustauluna. Tähän tauluun haettiin aloitustiedosta SELECT-lauseilla asiakkaan määritysten mukaiset tiedot. Tauluun haettiin kaikki yksilöivät ID-tiedot ID-sarakkeeseen aloitustiedosta asiakkaan migraatioehtojen mukaisesti ja näille ID-tiedoille pyöritettiin uudet ID-tiedot ID_NEW-sarakkeeseen.

MIGRATION_MAP-taulun TAULU-sarake ja MIGRATION_ID_VALUES-taulun ID-sarakkeita käytettiin relaatioina toisiinsa. Esimerkiksi MIGRATION_ID_VALUES-taulussa ID:llä 1 on kohdejärjestelmän asiakastaulu. MIGRATION_MAP-tauluun TAULU-sarakkeeseen haettiin tämän taulun tiedot eli MIGRATION_MAP-taulussa TAULU-sarake oli tällöin 1.

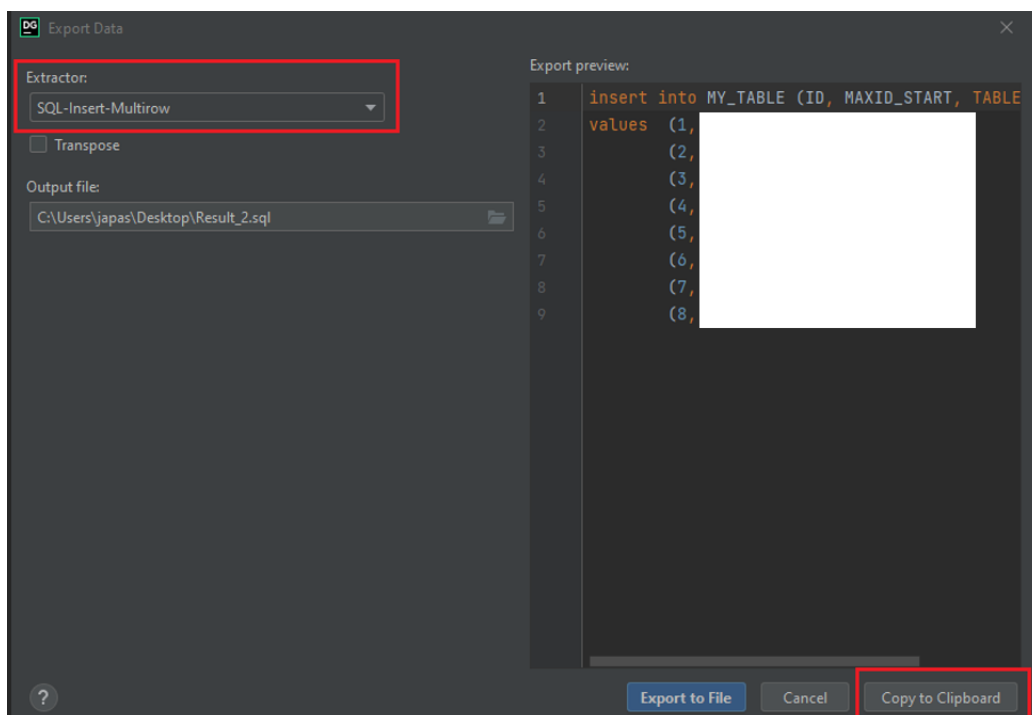
```

1  -- MIGRATION_ID_VALUES-taulun luontilause
2  CREATE TABLE IF NOT EXISTS `MIGRATION_ID_VALUES`
3  (
4      `ID`          bigint(20) NOT NULL,
5      `TABLE_NAME` varchar(255) NOT NULL,
6      `MAXID_START` bigint(20)          DEFAULT NULL,
7      `MAXID_END`   varchar(255)        DEFAULT NULL,
8      `UPDATED`     DATETIME           NOT NULL DEFAULT CURRENT_TIMESTAMP
9  );
10
11 -- MIGRATION_MAP-taulun luontilause
12 CREATE TABLE IF NOT EXISTS `MIGRATION_MAP`
13 (
14     `TAULU` INT(11) NULL DEFAULT NULL,
15     `ID`    INT(11) NULL DEFAULT NULL,
16     `ID_NEW` INT(11) NULL DEFAULT NULL,
17     INDEX `TAULU` (`TAULU`),
18     INDEX `ID` (`ID`),
19     INDEX `ID_NEW` (`ID_NEW`)
20 );

```

Kuva 17. Aloitusdatan tietokannan aputaulujen luontilauseet.

Aputaulujen luonnin jälkeen MIGRATION_VALUES_ID-tauluun haettiin omiksi riveikseen kohdejärjestelmän migraatiossa päivitettävien taulujen nimitiedot ja näiden yksilöivien tietojen maksimi arvot. Nämä otettiin DataGripin export data -ominaisuudella SQL-Insert-Multirow-muotoon ja ajettiin aloitusdatan tietokannan MIGRATION_VALUES_ID-tauluun.



Kuva 18. Export Data -ominaisuus, SQL-Insert-Multirow

Kun kohdejärjestelmän taulut oli tallennettu omiksi rivitiedoikseen MIGRATION_ID_VALUES-tauluun, pystyttiin aloittamaan MIGRATION_MAP-taulun kartoittaminen. Lähtödatan tietokantana muodostettiin jokaiselle datamigraatiossa käsiteltävälle taululle SELECT-lauseet kuvitteellisten asiakkaiden mukaisesti. SELECT-lauseissa kartoitettiin jokaisen taulun rivitiedot MIGRATION_MAP-sarakkeiden mukaisesti. MIGRATION_MAP-taulun TAULU-sarake määriteltiin käsiteltävän taulun mukaisesti, ID-sarakkeeseen valittiin lähtödatan yksilöivä tieto ja ID_NEW-sarakkeeseen pyöritettiin uudet ID-tiedot MIGRATION_ID_VALUES-taulusta MAXID_START-sarake + 1 mukaisesti. SELECT-lauseeseen lisättiin kaikki migraatiossa huomioitavat taulut UNION-funktiolla. Kun lause oli tehty migraatioehtojen mukaisesti kaikkien taulujen osalta, otettiin tulostuksesta Data Export -toiminnolla SQL-Insert-Multirow-lause, joka ajettiin lähtödatan MIGRATION_MAP-tauluun.

```
-- Tällä lauseella haetaan MIGRATION_ID_VALUES-taulun rivin mukainen taulu ID:llä, aloitusdatan vanha ID ja kohdejärjestelmään pyöritettävä uusi ID
SELECT *
FROM (
  (SELECT '1' AS TAULU,
         A.ASIKASID AS ID,
         (@RIV11 := @RIV11 + 1) AS ID_NEW
   FROM (SELECT ASIAKASID
         FROM ASIAKAS
         WHERE ASIAKASID IS NOT NULL
         GROUP BY ASIAKASID) A,
        (SELECT @RIV11 := (SELECT MAXID_START FROM MIGRATION_ID_VALUES WHERE ID = 1)) B
   ORDER BY A.ASIKASID)
  UNION
  (SELECT '2' AS TAULU,
         A.HENKILOID AS ID,
         (@RIV12 := @RIV12 + 1) AS ID_NEW
   FROM (SELECT HENKILOID
         FROM HENKILOT
         WHERE HENKILOID IS NOT NULL
         GROUP BY HENKILOID) A,
        (SELECT @RIV12 := (SELECT MAXID_START FROM MIGRATION_ID_VALUES WHERE ID = 2)) B
   ORDER BY A.HENKILOID)
) C;
```

Kuva 19. Esimerkki ASIAKAS- ja HENKILO-taulujen SELECT-lauseista

Kun tiedot oli ajettu MIGRATION_MAP-tauluun, voitiin aloittaa konversiolauseiden teko. Jokaiselle lähtöjärjestelmän taululle tehtiin yksilöllinen SELECT-lause, jossa annotoitiin lähtöjärjestelmän sarakkeelle kohdejärjestelmän sarakkeen nimi. Kohdejärjestelmän yksilöivälle riville annotoitiin MIGRATION_MAP-taulun yksilöivän sarakkeen tieto käyttämällä JOIN-funktiota, yhdistämällä lähtödatan taulun ID MIGRATION_MAP-taulun ID- ja TAULU-tietoon. Mikäli taulu sisälsi muiden taulujen yksilöiviä tietoja, tuli myös näille tehdä JOIN-funktio MIGRATION_MAP-taulun tietoihin, jotta myös näille saatiin kohdejärjestelmän uusi yksilöivä taulun tieto ja näin relaatio säilyi myös kohdejärjestelmässä oikeiden tietojen välillä. Tästä vaiheesta esimerkki kuvassa 20. Tässä vaiheessa oli myös hyödyllistä tarkistaa kohdejärjestelmän tauluihin vastaavat numerot MIGRATION_MAP-taulusta liittämällä MIGRATION_MAP-tauluun MIGRATION_ID_VALUES-taulu JOIN-funktiolla, tällä varmistettiin, että käsiteltiin oikean taulun numeroa. Tarkempi lause kuvassa 21.


```

-- ASIAKAS
SELECT MM.ID_NEW      AS -- kohdejärjestelmän yksilöivän sarakkeen tieto,
      MA.ID_NEW      AS -- kohdejärjestelmän tieto,
      /*
      SALASSA PIDETTÄVÄÄ TIETOA
      */
FROM ASIAKAS LA
JOIN MIGRATION_MAP MM ON (LA.ASIKASID = MM.ID AND MM.TAULU = 1) -- tietojen kartoitukset JOIN-funktiolla
LEFT JOIN MIGRATION_MAP MA ON (LA.HENKILOID = MA.ID AND MA.TAULU = 2);

```

Kuva 20. Tietojen kartoittaminen

```

-- lause, jolla tarkistetaan taulujen numerot TAULU-sarakeesta migration_map-tilusta
SELECT MM.TAULU, MIV.TABLE_NAME, MIV.MAXID_START, MIV.MAXID_END FROM MIGRATION_MAP MM
JOIN MIGRATION_ID_VALUES MIV ON MIV.ID = MM.TAULU
GROUP BY TAULU;

```

Kuva 21. Kohdejärjestelmän taulujen numeroinnin tarkistuslause konversiolauseita varten

Kun tiedot oli kartoitettu ja SELECT-lause muodostettu, otettiin Export Data -toiminnolla SQL-Insert-multirow-lause, joka ajettiin kohdejärjestelmään. Tämä tietojen kartoitus toteutettiin jokaiselle taululle. Jokaisen taulun konversiolauseiden ajon jälkeen kohdejärjestelmään, otettiin kohdejärjestelmästä taulun uusi maksimi ID MAX()-funktiolla ja päivitettiin tämä MIGRATION_ID_VALUES-tiluun. Näin pystyttiin tarkastamaan heti konversiolauseiden sisäänajon jälkeen, että haluttu rivimäärä oli ajettu sisään onnistuneesti. Kaikkien konversiolauseiden kohdejärjestelmän sisäänajon jälkeen voitiin tarkistaa relaatiot ja kohdejärjestelmän toiminnot kohdejärjestelmän käyttöjärjestelmästä ja todeta datamigraatio työn onnistuminen. Tarkastus toteutettiin sattumanvaraisesti useammalla eri tiedolla datamigraatiossa käsitellyille tauluille.

4 TULOKSET

Opinnäytetyön tavoitteena oli toteuttaa perehdytysmateriaali tilaajalle ja kouluttaa opinnäytetyön tekijä datamigraatio työhön. Perehdytysmateriaalin tuli toteuttaa aloitusdata ja tehdä tälle datalle datamigraatio tilaajan toiminnanohjausjärjestelmään, sekä dokumentoida tämän datamigraation prosessi, selittää prosessin vaiheet ja tärkeimmät toiminnot ja työkalut.

Koska kohdejärjestelmänä tässä perehdytysmateriaalin datamigraatiossa toimi hyvin räätälöitävä toiminnanohjausjärjestelmä ja järjestelmässä voidaan usealla valinnalla vaikuttaa kohdejärjestelmän lopullisen datan muotoon, on jokainen datamigraatio tähän järjestelmään yksilöllinen. Tämän lisäksi aloitusdata tuli toteuttaa mahdollisimman geneeriseksi, eikä tässä voitu käyttää asiakasdataa tietoturvasyistä. Nämä edellä mainitut asiat vaikeuttivat perehdytysmateriaalin - ja datamigraation toteutusta materiaaliin. Perehdytysmateriaalissa ja aloitusdatassa on tästä huolimatta esitetty usein muokattavia tietoja ja tietotyyppejä, ja tärkeimpänä ohjeistettu selkeästi, kuinka tilaajan datamigraatiotyössä tietojen kartoituksen avulla voidaan lähtöjärjestelmän datasta muokata konversiolauseet ja datan kartoituksen avulla siirtää data järjestelmien välillä. Datan kartoitus on tärkeä ymmärtää, sillä se luo perustan tilaajan datamigraatiotyön toteutukseen ja tämän avulla voidaan toteuttaa toiminnanhallintajärjestelmään asiakaskohtaisia muutoksia.

5 POHDINTA

Mielestäni opinnäytetyölle asetetut tavoitteet saavutettiin hyvin. Perehdytysmateriaalista saatiin selkeä kokonaisuus, opinnäytetyön tekijä oppi ymmärtämään datamigraatiotyön perusteet niin teorian - , kuin tilaajan toteutuksen osalta ja perehdytysmateriaali voidaan ottaa tilaajan käyttöön jatkossa datamigraatiotyön koulutuksessa.

Työn teknillisessä osassa tuli vastaan monenlaisia haasteita, kuten aloitusdataan valitun työkalun Mockaroo -sivuston muutokset työn toteutuksen aikana. Lisäksi aloitusdatan tuli olla geneerinen ja konversiolauseiden muodostamisessa haasteita tuotti kohdejärjestelmän suuruus, tietokantataulujen pakollisuudet ja niiden useat relaatiot tauluihin, joita ei ollut toteutettu aloitusdatassa. Näistä haasteellisinta oli toteuttaa konversiolauseet geneeriselle suppealle aloitusdatalle, jotta data saataisiin validiin muotoon kohdejärjestelmässä. Lopulta saatiin muodostettua toimivat ratkaisut, jotta perehdytysmateriaaliin saatiin tarvittava ohjeistus ja materiaali datamigraatiotyön toteutukseen.

Koska perehdytysmateriaalissa käytettiin opinnäytetyön tekijän muodostamaa geneeristä CRM dataa salassa pitovelvollisuuden takia, voisi jatkokehitysmielessä tilaajan materiaaliin vaihtaa aloitusdatan vastaamaan jotain jo tilaajan toteuttamaa datamigraatiota. Toki tämä data tulisi anonymisoida, jotta asiakasdataa voitaisiin käyttää Euroopan GDPR säädösten mukaisesti. Näin ollen saataisiin validia dataa siirrettäväksi datamigraatiotyöhön, joka voisi helpottaa perehdytettävää visualisoimaan kohdejärjestelmästä siirretyn datan. Muuten perehdytysmateriaalia voitaisiin käyttää sellaisenaan ja se toimii jo sellaisenaan hyvänä oppaana ymmärtämään tietojen kartoittamisen ja toteuttamisen.

Tämän työn selvitykset ja toteutuksen tulokset, jotka opinnäytetyössä esitetään, ovat hyödyllisiä datamigraatioiden parissa työskenteleville, jotka etsivät ratkaisuja tietojen manuaaliseen kartoittamiseen ja konversiolauseiden toteuttamiseen hyödyntäen tietojen kartoittamista ja aputauluja.

6 LÄHDELUETTELO

JetBrains. (ei pvm). *Datagrip*. Haettu 10. 12. 2021 osoitteesta jetbrains.com:

<https://www.jetbrains.com/datagrip/>

Katkamwar, R. (15. 04. 2016). *How to ensure a succesful data migration 6 critical validation tests*.

(InfoCepts) Haettu 6. 01. 2022 osoitteesta InfoCepts.com: <https://www.infocepts.com/blog/how-to-ensure-a-successful-data-migration-6-critical-validation-tests/>

NetApp. (ei pvm). *What is data migration*. Haettu 09. 12. 2021 osoitteesta netapp.com:

<https://www.netapp.com/knowledge-center/what-is-data-migration/>

Pinja Oy. (ei pvm). *Pinja*. Haettu 19. 11. 2021 osoitteesta pinja.com: <https://pinja.com/pinja>

Prohorchik, K. (22. 04. 2021). *Data migration strategy*. (Itransition) Haettu 13. 01. 2022 osoitteesta

itransition.com: <https://www.itransition.com/blog/data-migration-strategy>

STTK. (ei pvm). *Perehdytys*. Haettu 26. 11. 2021 osoitteesta tyoelamaan.fi:

<https://tyoelamaan.fi/tyosuhde/perehdytys/>

Talend. (ei pvm). *Understanding data migration strategies best practises*. Haettu 09. 12. 2021 osoitteesta

talend.com: <https://www.talend.com/resources/understanding-data-migration-strategies-best-practices/>

Talend. (ei pvm). *What is Data Mapping?* Haettu 15. 02. 2022 osoitteesta

<https://www.talend.com/resources/data-mapping/>

Tietosanakirja-titanica. (2022). *Encyclopedia-titanica*. Haettu 6. 01. 2022 osoitteesta Significado de ad hoc:

<https://fi.encyclopedia-titanica.com/significado-de-ad-hoc>

Työturvallisuus-keskus. (ei pvm). *Perehdyttaminen ja tyonopastus - ennakoivaa tyosuojelua*. Haettu 15. 12.

2021 osoitteesta ttk.fi: https://ttk.fi/oppaat_ja_ohjeet/digijulkaisut/perehdyttaminen_ja_tyonopastus_-_ennakoivaa_tyosuojelua

Warmuth A., D. B. (2015). *DS8870 Data Migration Techniques* (Ensimmäinen painos p.). IBM Redbooks.

Haettu 17. 12. 2021 osoitteesta

<https://play.google.com/books/reader?id=3fHDCAAAQBAJ&pg=GBS.PP1&hl=en>