Informaatioteknologian tiedekunnan julkaisuja
No. 65/2018

Henri Heinonen, Pekka Neittaanmäki, Teemu Hyytiäinen, Alvar Mahlberg, Kasper Tontti ja
Ville Yli-Pelkonen

Älysopimusohjelmointi sosiaali- ja terveysalalla

JYVÄSKYLÄN YLIOPISTO

Informaatioteknologian tiedekunnan julkaisuja
No. 65/2018

Editor: Pekka Neittaanmäki
Covers: Petri Vähäkainu ja Matti Savonen

ISBN 978-951-39-7645-3 (verkkoj.)
ISSN 2323-5004

Jyväskylä 2018

Älysopimusohjelmointi sosiaali- ja terveysalalla

Henri Heinonen
Pekka Neittaanmäki
Teemu Hyytiäinen
Ville Yli-Pelkonen
Alvar Mahlberg
Kasper Tontti

Tämä julkaisu on toteutettu osana WHC-hanketta, johon Jyväskylän yliopisto on saanut rahoituksen Business-Finlandilta.

Business Finland-hanke: WHC

KUVIOT

SISÄLLYSLUETTELO

1 Johdanto

Lohkoketjuteknologia mahdollistaa useiden toisiinsa luottamattomien tahojen luoda yhteinen tilikirja tai tietokanta ilman luotettua kolmatta osapuolta. Näiden ominaisuuksiensa ansiosta se on saanut huomattavasti julkisuutta ja sen erilaisia sovellusvaihtoehtoja pohditaan laajalti muun muassa varallisuuden siirroissa, tuotteiden kiertokulun seuraamisessa, esineiden internetissä ja terveydenhuollossa. Lohkoketjuteknologia mahdollistaa hajautetun tilikirjan ja hajautetun tietokannan. Se koostuu useista datapaketeista, eli lohkoista, jotka linkitetään toisiinsa tiivisteiden avulla. Linkitetyistä lohkoista syntyy lohkoketju, jonne lohkoketjun arkkitehtuurista riippuen, voivat kaikki, tai ennalta valittu ryhmä, kirjoittaa ja lukea dataa. Tiedon oikeellisuus varmistetaan kirjoitushetkellä erilaisilla konsensusmekanismeilla, jonka jälkeen tieto tallennetaan lohkoketjuun. Tiedon tallentamisen jälkeen sen muuttaminen on hankalaa ja tämän ominaisuuden ansiosta lohkoketjuteknologia sopii useiden osapuolten yhteiseksi tilikirjaksi tai tietokannaksi, sillä osapuolten ei tarvitse luottaa toisiinsa (Wüst & Gervais, 2017). Järjestelmä pitää huolen, etteivät osapuolet pääse muuttamaan dataa.

Sosiaali- ja terveysalalla on laajasti erilaisia toimijoita ja heillä kaikilla on omat tietokantansa. Tämän seurauksena kommunikointi eri toimijoiden välillä ei ole automaattista. Jos tietokannat avattaisiin kaikkien käytettäväksi, heräisi useita kysymyksiä, kuten kuka vastaa mistäkin tietokannasta ja kenellä on oikeus kirjoittaa tietokantaan uutta dataa. Lohkoketju antaa näihin kysymyksiin yksinkertaisen vastauksen, sillä sen ylläpitäminen olisi kaikkien siinä toimivien toimijoiden vastuulla ja kaikilla lohkoketjua ylläpitävillä tahoilla tulee olla sinne kirjoitusoikeus. Tämän ansiosta kaikilla toimijoilla olisi myös jatkuvasti kaikki tarvittava data saatavilla luotettavasti.

Seuraavassa kappaleessa käydään läpi lohkoketju ja sen tuomat mahdollisuudet ja haasteet. Kappaleessa 3 kerrotaan, mitä älysopimukset ovat ja miten ne ovat yhteydessä lohkoketjuihin, minkä jälkeen kappaleessa 4 pohditaan älysopimusten ja lohkoketjujen käyttöä sosiaali- ja terveysalalla. Kappale 5 keskittyy älysopimusohjelmointiin ja demonstraatioon lohkoketjun päällä toimivasta palvelusetelialustasta. Viimeinen kappale sisältää yhteenvedon läpikäydyistä asioista.

2 Lohkoketjuteknologia

2.1 Lohkoketjuteknologia yleisesti

Lohkoketju koostuu ketjusta, jossa on useita peräkkäisiä datapaketteja. Näitä datapaketteja kutsutaan lohkoiksi. Yksittäinen lohko sisältää useita tapahtumia. Tapahtumien lisäksi lohkoketjuun voidaan säilöä kuvia, tekstiä tai koodia. Lohkoketju laajentuu aina uudella lohkolla ja sen seurauksena lohkoketju edustaa kokonaista tilikirjaa kaikista tapahtumista (Nofer ym., 2017). Koska lohkoketjuun voidaan tallentaa muutakin dataa kuin tapahtumia, se myös toimii tietokantana. Se koostuu muuttumattomista, järjestyksessä toisiinsa linkitetyistä datalohkoista. Lohkojen ketjuksi liittämisessä käytetään hyväksi lohkojen tiivisteitä. Jokaisen lohkon tietosisällöstä lasketaan yksilöllinen tiiviste. Tähän tietosisältöön lisätään myös edellisen lohkon tiiviste. Kun kaikissa lohkoissa lukee edellisen lohkon tiiviste, linkittyvät nämä lohkot yhdeksi ketjuksi (Storås, 2016). Tämä tarkoittaa sitä, että jos yksikin merkki jossain lohkossa muuttuu, vaikuttaa se kaikkiin sen jälkeisiin tiivisteisiin. Tämän ansiosta lohkoketjun tapahtumahistoriaa on erittäin vaikea muuttaa jälkikäteen. Lohkoketjuun tallennettu tieto on hajautettu siten, että se sijaitsee fyysisesti ja digitaalisesti samaan aikaan useassa eri paikassa, niin sanotuissa verkon solmukohtana toimivissa tietokoneissa. Nämä solmukohdat ovat lohkoketjua ylläpitäviä tietokoneita. Näin varmistetaan tiedon saatavuus, säilyvyys ja käytettävyys. Lohkoketjuissa tieto on tallennettu useaan eri sijaintiin ja uuden tiedon lisääminen validoidaan ja varmistetaan kussakin paikassa. Tämän seurauksena lohkoketjun käyttö ei vaadi luottamusta käyttäjiensä tai kolmansien osapuolten välillä. ”Lohkoketju on parhaimmillaan silloin, kun osapuolilla on eriävät intressit, eivätkä he voi täysin luottaa toisiinsa”, kuvailee Pekka Nikander (Kotilainen, 2017).

Lohkoketjuja on periaatteessa kolmea eri tyyppiä: julkisia, yksityisiä ja konsortiolohkoketjuja (Zheng ym., 2017). Bitcoin on esimerkki julkisesta lohkoketjusta. Siinä tarkoituksena on päästä eroon maksuliikenteen välikäsistä: jokainen verkon solmukone tallentaa koko lohkoketjun ja varmistaa kaikki transaktiot. Hyvä puoli on, että kuka tahansa pääsee lukemaan ja lähettämään transaktioita, mutta ongelmaksi muodostuu verkon hitaus ja transaktioiden kohonneet siirtokulut. Yksityisessä lohkoketjussa yritys toimii välikätenä, joka varmistaa kaikki transaktiot tehostaen verkon toimintaa: koko verkon ei tarvitse osallistua, joten siirtäminen on nopeampaa ja halvempaa. Ongelmana on se, että on luotettava yrityksen kykyyn hallita lohkoketjun tietoturvaa (Thompson, 2018). Konsortiolohkoketju on osittain yksityinen. Tällaisessa ratkaisussa on annettu esimerkiksi muutamalle taholle lupa osallistua transaktioiden verifiointiin. Tämä ratkaisumalli voi olla myös yhtä nopea kuin puhtaasti yksityinen lohkoketju ilman, että kaikki hallinto keskittyy yhdelle yritykselle. (Thompson, 2018).

Vaikka lohkoketjut ovat uutta teknologiaa, on niiden käyttöönotto edullista. Lohkoketjusovellukset rakentuvat olemassa olevan digitaalisen datan ja ICT-infrastruktuurin päälle, mikä alentaa kokeilemisen kustannuksia ja mahdollistaa uusien käyttötapausten ilmestymisen nopeasti (Lansiti & Lakhani, 2017).

2.2 Lohkoketjuteknologian määrittely

Lohkoketjuteknologia nähdään yleisesti osana hajautetun tilikirjan (engl. Distributed ledger technology) ratkaisuja. Hajautettu tilikirja on erityisesti omaisuustietokanta, joka voidaan jakaa verkossa eri toimijoiden kesken hajautetusti. Hajautetun tilikirjan ratkaisuja voidaan toteuttaa myös ilman varsinaista lohkoketjua. (Walport, 2016.) Lohkoketjut voidaan jakaa kolmeen ryhmään: julkiset lohkoketjut, yksityiset lohkoketjut ja konsortiolohkoketjut. Lohkoketjujaottelu tapahtuu tarkemmin seuraavasti:

1. **Julkiset lohkoketjut**: ovat avoimia kaikille, tarjoavat läpinäkyvyyttä ja pyrkivät estämään vallan keskittymistä tietylle yksittäiselle taholle. Kaikki toimijat voivat osallistua järjestelmän ylläpitoon, käyttämiseen ja transaktioiden hyväksymiseen. (Wüst & Gervais, 2017). Julkisessa lohkoketjussa toimijoiden ei tarvitse luottaa yksittäiseen tahoon, vaan toimijat luottavat, että suurin osa järjestelmän ylläpitäjistä haluavat järjestelmän toimivan tarkoituksenmukaisesti. Järjestelmässä ei ole kannattavaa toimia tarkoituksenmukaisuutta vastaan, sillä tarkoituksenmukaiseen toimintaan kannustetaan erilaisilla taloudellisilla kannustimilla (AnythingCrypto, 2018), esimerkiksi louhintapalkkiolla, jonka toimija saa uuden lohkon löydyttyä. Julkisessa lohkoketjussa kuka tahansa voi tarkastella lohkoketjun sisältöä ja osallistua sen ylläpitämiseen louhijana tai solmuna. Vaikka lohkoketjun sisältö onkin julkisesti tarkasteltavissa, se voi silti olla salattua, jolloin vain salausavaimen hallussapitäjät voivat muuntaa sen ihmiselle luettavaan muotoon. (Wüst & Gervais, 2017).

Julkisessa lohkoketjussa lohkoketjun luojalla on verrattain vähän valtaa lohkoketjun käyttäjiin verrattuna muihin lohkoketjutyyppeihin. Jos julkisen lohkoketjun kehittäjä haluaa tehdä niin sanotun kovan haarautuksen (engl. hard fork) merkittävällä protokollamuutoksella, solmukohdat voivat valita olla implementoimatta näitä muutoksia, jolloin lohkoketju haarautuu; protokollamuutoksen implementoineet solmut toimivat lohkoketjun eri haarassa kuin protokollamuutoksen implementoimattomat solmut (Lewis, 2015). Kummatkin haarat ovat tässä tapauksessa aivan yhtä valideja. Julkinen lohkoketju on paras vaihtoehto esimerkiksi kryptovaluuttaa käsitteleville lohkoketjuille ja arvonsiirtojärjestelmille, joihin ei haluta käyttäjien välille välittäjää tai välimiestä.

2. **Yksityiset lohkoketjut**: Yksityiset lohkoketjut toimivat suljetummissa ympäristöissä ja käyttäjillä on useasti tiettyjä rajattuja oikeuksia. Jokaisella halukkaalla ei ole lupaa osallistua lohkoketjun käyttöön. Yksityinen lohkoketju soveltuu paremmin tietyille toimialoille, joissa halutaan toimia ainoastaan luotettujen kumppaneiden kanssa. Yksityinen lohkoketju on keskitetty yhden vahvasti luotetun toimijan hallinnoimaksi, joka voi tällöin tehokkaasti muokata lohkoketjun toimintaperiaatteita, päällekirjoittaa ja korjata haluamiaan transaktioita ja lohkoketjun syötteitä, sekä poistaa tai lisätä lohkoketjun lukuoikeudellisia toimijoita (Wüst & Gervais, 2017). Keskitetyn kirjoitusoikeuden vuoksi mahdolliset solmujen väärinkäytöstä johtuvat hyökkäykset ovat äärimmäisen epätodennäköisiä, sillä yksityisessä lohkoketjussa yhdellä taholla on oikeudet ja mahdollisuus muuttaa dataa parhaaksi kokemallaan tavalla. Tästä seuraa tahon varsinaisesta

luotettavuudesta riippuen joko erittäin negatiivisia tai positiivisia implikaatioita yksityisyyteen, tiedon varmentamiseen, tarjotun tiedon oikeellisuuteen, muuttumattomuuteen sekä lohkoketjun toiminnan jatkuvuuteen. Yksityisen lohkoketjun hallinnoijan on mahdollista väärinkäyttää lohkoketjua.

Yksityisessä lohkoketjussa on syytä miettiä tarkkaan, mille taholle tulisi sallia keskitetyt kirjoitusoikeudet, ja onko lohkoketjun käytöstä tässä tapauksessa lainkaan hyötyä — lohkoketjun toiminta voitaisiin toteuttaa keskitetysti ja/tai jo olemassa olevilla, hyviksi todetuilla ja todistetuilla tekniikoilla. Lohkoketjut ovat mielekkäimpiä ratkaisuja tilanteissa, jossa kaikki keskenään kommunikoivat osapuolet ovat keskenään luottamuksettomia tai matalan luottamuksen toimijoita; muussa tapauksessa olisi halvempaa ja mahdollisesti toimivampaa toteuttaa lohkoketjutoiminnallisuus perinteisillä tietokannoilla ja niiden yhteyteen ohjelmoitavilla rajapinnoilla. Yksityinen lohkoketju on paras vaihtoehto muutamille käyttötapauksille yksittäisten yhtiöiden sisäisen toiminnan mahdollistamiseksi.

3. **Konsortiolohkoketjuja:** kutsutaan myös nimellä hybridi-lohkoketju (Buterin, 2015). Konsortiolohkoketju sijoittuu ominaisuuksiltaan julkisen ja yksityisen lohkoketjun väliin. Konsortiolohkoketjut ovat soveltuvia järjestelmiin, joissa toimijoiden välillä tulee olla jonkinlaista luottamusta. (Zheng ym., 2017.) Konsortiolohkoketju on valikoitujen toimijoiden ylläpitämä lohkoketju, mutta sen sisältö voidaan asettaa kenen tahansa näkyviin, tai sitä voidaan rajata näkyväksi vain sallituille osapuolille ja lohkoketjuun tunnistautuneille. Konsortiolohkoketju mahdollistaa tiukemman oikeuksienhallinnan rajatun käyttäjäkunnan kesken tarjoten silti lohkoketjun rajatun luottamuksettomuuden, autonomisen ja automaattisuuden ja muuttumattomuuden tuomat edut lohkoketjun käyttäjien piiriin. Konsortiolohkoketjun rajatummasta kirjoitusoikeuksellisesta käyttäjäkunnasta johtuen teoreettiset maksiminopeudet ovat marginaalisesti pienemmät, ja hajautuneisuuden aste on matalampi. Konsortiolohkoketjun solmujen voidaan kuitenkin luottaa olevan tehokkaita ja hyvin laajalle verkostoituneita, jolloin relatiivinen nopeus on todennäköisesti suurempi kuin suurissa julkisissa lohkoketjuissa. (Zheng ym., 2017.)

Konsortiolohkoketju on paras vaihtoehto kahden tai useamman organisaation yhteiseksi tietojärjestelmäksi. Esimerkiksi Kela, sairaanhoitopiirit, vakuutusyhtiöt ja lääkeyhtiöt voivat muodostaa yhdessä lohkoketjun, jonka solmuina kukin toimisi. Kirjoitusoikeudet on rajattu näille toimijoille, ja konsensusmekanismi ei vaadi erillistä taloudellista kannustinta lohko- ja louhintapalkkioiden muodossa — lohkoketjussa toimijat korvaisivat tässä tapauksessa osan omasta tietoteknisestä infrastruktuuristaan lohkoketjulla, jolloin olisi heidän oman etunsa mukaista toimia lohkoketjussa oikein. Tietojen näkeminen voitaisiin rajata, jolloin potilaat näkisivät lohkoketjusta vain omia tietojaan tunnistautumisen jälkeen.

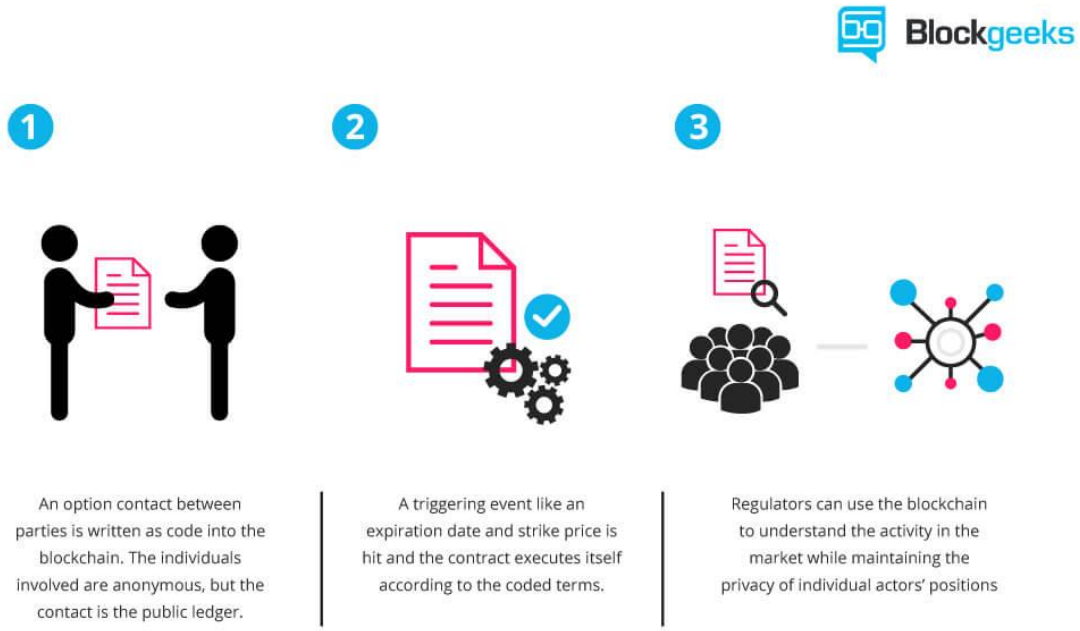
2.3 Lohkoketjuteknologian haasteet ja mahdollisuudet

Lohkoketjuille on tyypillistä, että niihin voidaan lisätä tietoa, mutta poistaminen ja tiedon muokkaaminen eivät ole mahdollisia. Virheellinen tieto lohkoketjussa voidaan korjata tekemällä uusi merkintä, mutta alkuperäisen (eli tässä tapauksessa virheellistä merkintää) on erittäin vaikea muuttaa jälkikäteen. (Lewis, 2016.) Esimerkiksi rahasiirroissa tämä on toivottava ominaisuus, koska usein halutaan saada varmuus rahan alkuperästä. Sosiaali- ja terveysalalla tämä luo haasteita, koska Euroopan unionin GDPR-asetus vaatii, että asiakkailla on oikeus pystyä poistamaan järjestelmistä heitä koskevat tiedot. Tämä voidaan kuitenkin kiertää sillä, että kaikki yksilöivät tiedot säilytettäisiin perinteisillä palvelimilla, jotka integroidaan osaksi lohkoketjua.

Lohkoketju voisi tarjota useita erilaisia hyötyjä sosiaali- ja terveysalalle, sillä se mahdollistaa hajautetun tilikirjan pitämisen, jossa kuka tahansa voi lukea ja kirjoittaa lohkoketjuun. Tämän toiminnan ansiosta asiakkaan tiedonsiirto ja maksuliikenne voitaisiin toteuttaa lohkoketjun ja älysopimuksien avulla huomattavasti automaattisemmin kuin nykyään (Enisa, 2017).

3 Älysopimukset yleisesti

Älysopimus on lohkoketjussa ehtojen täytyttyä itsensä toteuttava tietokoneohjelma. Älysopimukset poikkeavat tekojen, puheen tai kirjoituksen avulla syntyvistä tavanomaisista sopimuksista, sillä ne sijaitsevat ohjelmistossa ja ovat itseään toteuttavia. Älysopimusten odotetaan mahdollistavan liiketoimintojen prosessiautomaation huomattavasti aiempaa laajemmassa mittakaavassa. Lainsäädännön näkökulmasta älysopimusten synnyttämien oikeussuhteiden pätevyys on vielä epäselvää (Lauslahti ym., 2016). Älysopimus tekee mahdolliseksi vahvistaa sopimusosapuolten suorittaneen velvoitteensa, ja se mahdollistaa nopeutetun ja automatisoidun suorituksen vaadittujen ehtojen täyttyessä (Enisa, 2017). Kuviossa 1 näkyy, kuinka älysopimus luodaan kahden osapuolen välillä. Älysopimus kirjataan lohkoketjuun ja siihen määritellään sopimuksen ehdot. Kun sopimuksen ehdot täyttyvät, kaikki sopimuksen osapuolet saavat sopimuksen mukaisesti osuutensa. Sopimukseen voidaan myös kirjata viimeinen päivä, jolloin sopimuksen ehtojen pitää täyttyä tai se purkautuu. Sopimuksen purkautuessa kaikille osapuolille palautetaan heidän osuutensa. Sopimuksen tekohetkestä alkaen regulaattorit voivat seurata sopimusten oikeellisuutta ja toimintaa ilman, että yksityisten toimijoiden henkilöllisyyttä tarvitsee paljastaa.



KUVIO 1. Älysopimusten toiminta (Blockgeeks)

Esimerkki: Älysopimuksen avulla toteutettu älykäs lukitus. Saksalainen startup-yritys slock.it yhdistää fyysiset objektit ja lohkoketjun älysopimukset. Heidän demonstroimassaan esimerkissä lukot voidaan avata ja lukita älysopimuksen avulla. Tällainen ratkaisu lisäisi esimerkiksi vertaisvuokrauksen luotettavuutta, ja se mahdollistaa jakamistalouden hajauttamisen ja toteuttamisen ilman luotettua kolmatta osapuolta. Hajautettu jakamistalous antaa kenelle tahansa mahdollisuuden helposti vuokrata, jakaa tai myydä mitä tahansa, joka voidaan lukita. (Kinnunen ym., 2017.)

4 Pohdintaa sosiaali- ja terveysalan mahdollisista sovelluksista

Lohkoketjuille on hahmotettu käyttömahdollisuuksia sosiaali- ja terveydenhuollossa. Esimerkiksi palveluiden rahaliikennettä voitaisiin automatisoida. Älysopimuksen tekevät automaattisesti niihin ohjelmoituja tehtäviä lohkoketjuissa (Enisa, 2017). Näiden avulla voidaan useita hallinnollisia tehtäviä automatisoida sosiaali- ja terveysalalla. Esimerkiksi, asiakkaalle voitaisiin antaa arvoa, jonka hän voi vaihtaa tarvitsemaansa sosiaali- tai terveydenhuollon palveluun minkä asiakas haluaa. Asiakkaan käyttäessä sosiaali- tai terveydenhuollon palveluita arvo siirtyisi automaattisesti palveluntarjoajalle, jolta arvo siirtyisi automaattisesti palvelunmaksajalle, kun asiakas on saanut palvelunsa. Kun palvelunmaksaja saa arvon, hän näkee lohkoketjusta, että asiakas on saanut oikean palvelun ja tämän jälkeen maksaa palveluntarjoajalle palvelusta. Näin palveluntarjoaja saisi automaattisesti maksun tuottamastaan palvelusta. Arvon siirtyessä palvelunmaksajalle voidaan arvo tuhota ja seuraavan kerran asiakkaan tarvitessa sosiaali- tai terveyspalveluita, voidaan sama toiminta toteuttaa uudestaan. Sote-palveluiden käytön volyymeissä palvelusetelien osuus oli 2016 alle yksi prosentti. Tieran palvelusetelijärjestelmän ylläpitoon ja tietojenkäsittelyyn Oulussa vaaditaan tällä hetkellä 12 henkilön työpanos vuodessa (Salonen ym., 2018).

Toinen mahdollinen sovellus sosiaali- ja terveysalalla voisi liittyä lääkejakeluketjujen kiertokulun seurantaan. Perinteisesti tuotteiden valmistus, kuljetus ja jälleenmyynti on tapahtunut eri toimijoiden toimesta ja tämän seurauksena tiedot ovat yksittäisissä, erillisissä järjestelmissä. Lohkoketjun avulla kaikki tiedot voitaisiin kirjoittaa samaan järjestelmään ja kuka tahansa voisi QR-koodin lukemalla nähdä koko tuotteen elinkaaren (Tian, 2018). Lääkkeiden elinkaarta lohkoketjussa seuraamalla voitaisiin estää väärennettyjen ja vanhentuneiden lääkkeiden myynti. Lisäksi kuluttajat pystyisivät halutessaan tekemään eettisempiä valintoja, jos olisi tiedossa minkälaisia kokeita kehitykseen on käytetty ja paljonko lääkkeen valmistus kuormittaa ympäristöä. Samankaltaista lohkoketjuratkaisua voitaisiin hyödyntää myös muita terveysalan laitteiden ja välineiden koko elinkaarta seuraavaan toimintaan, mikä parantaisi potilasturvallisuutta ja saattaisi alentaa kustannuksia. (Chapron, 2017.)

5 Palveluseteli-prototyyppi

5.1 Palveluseteli-prototyyppi yleisesti

Nykyisin sosiaali- ja terveysalalla on laaja kenttä erilaisia toimijoita. Jokaisella toimijalla on omat tarpeet ohjelmistonsa suhteen ja harvoin ohjelmistot ovat yhteensopivia toistensa kanssa. Jos tietokannat olisivat yhteisiä, kuka niistä vastaisi, miten tietokantoja saisi käyttää ja kuka määrittää oikeudet käyttää tietokantaa. Perinteisin keinoin keskitetyillä palvelimilla toteutettuna monen toimijan yhteinen tietokanta herättää monia kysymyksiä, joihin lohkoketju kykenee vastaamaan: jokainen osapuoli ylläpitää palvelua omalta osaltaan, oikeudet määritellään suhteilla ja suhteet määritellään älysopimuksin.

Esittelemämme palveluseteliprototyyppi on esimerkki siitä, että älysopimuksilla voidaan rakentaa yhteinen ympäristö erilaisille toimijoille. Se avaa saman rajapinnan sekä potilaalle, lääkärille, että yrityksille palvelusetelin tarkasteluun. Se ei ota kantaa, minkälainen on asiakkaan pääte: yrityksellä voi olla oma näkymänsä sopimuksiin ja niiden hallintaan ja potilas käyttää kunnan päätettä, tai kaikki käyttävät samaa päätettä. Älysopimukset pitävät huolen siitä, että rajapinta näyttää vain sen informaation, minkä asiakkaan tulee nähdä.

Lähdetään siitä oletuksesta, että jokaiselle ihmiselle on määritelty oma avain, jota kukaan muu ei tiedä. Tämä avain voi olla jokin tunnusluku tai se voi olla koodattuna Kela-korttiin tai mahdollisesti jonkinlaiseen USB-massamuistiin, jota ihmiset kantavat tulevaisuudessa mukanaan aina, tai että jokaisella ihminen on siru ja hänet voidaan tunnistaa sirusta. Henkilö voidaan siis tunnistaa palvelusetelipalvelussa omalla avaimellaan ja hän voi tarkastella omia tietojaan kirjautumatta erikseen sisään. Skenaariossa on olemassa tietokannat, josta palvelusetelirajapinta voi hakea tarvittavat tiedot älysopimusten määrittämien sääntöjen mukaan.

Toteutettu prototyyppi simuloi tätä skenaariota käyttäen yksityistä Ethereum-älysopimusalustan päälle rakennettua verkkoa sekä Metamask-nimistä Ethereum-lompakko-ohjelmaa. Metamask säilyttää käyttäjän yksityisen avaimen ja tätä avainta käytetään henkilön tunnistamiseen. Simulaation aikana avaimia voidaan yhdistää (kuvitteelliseen) henkilöllisyyteen. Simulaatiossa tietokannat luodaan alussa ja sen jälkeen vain palvelusetelitietokantaan kirjoitetaan.

Oletetaan, että kunta voi myöntää rajattomasti palveluseteleitä ja lääkäri voi omalla harkinnallaan määrätä minkä tahansa arvoisen palvelusetelin potilaalle. Palveluseteli on kerran käytettävä ja arvo on muuttumaton. Simulaatio ei ota myöskään kantaa siihen, miten käytännössä seteleihin sidotaan arvo ja miten arvo siirtyy yritykselle. Yritykset tuottavat kaikkia mahdollisia terveydenhuollon palveluita, eikä palvelun todellista tuottamista tarvitse todentaa.

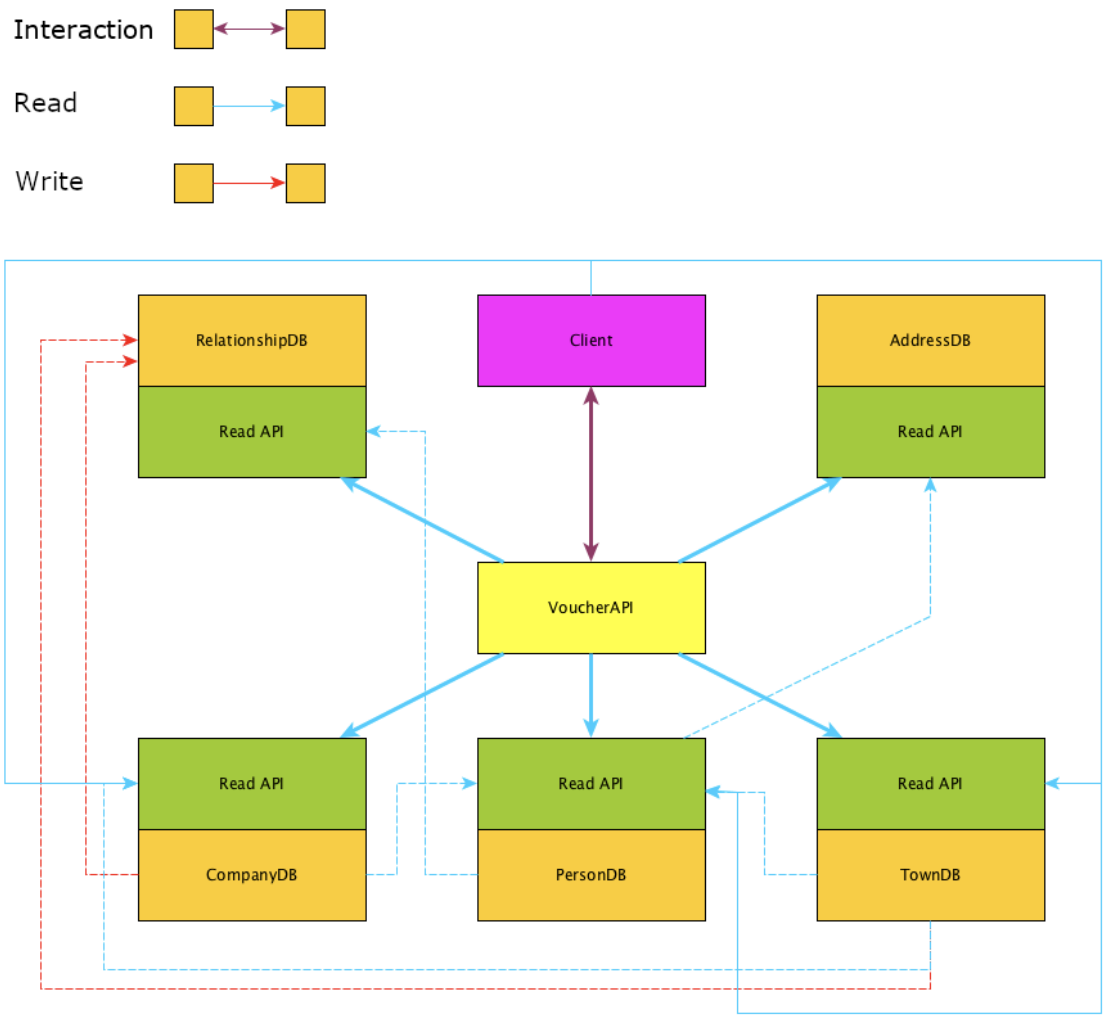
Simulaatiossa on kolme palvelua käyttävää osapuolta: kunnan lääkärit, sote-yrityksen työntekijät ja potilaat. Lääkärit voivat myöntää kunnan puolesta palveluseteleitä potilaille ja yrityksen työntekijät voivat sitoutua tuottamaan palveluita potilaalle ja palvelun tuottamisen jälkeen lunastaa setelin arvon.

5.2 Voucher API-älysopimukset

5.2.1 Kokonaisuus

Palveluseteli-prototyyppi on kokonaisuus, joka mallintaa skenaariota, jossa kaikki data on tallennettu lohkoketjuihin. Tarvitaan siis tietokannat, joista haetaan tiedot henkilöistä, yrityksistä ja kunnista. Tarvittavat tietokannat on ohjelmoitu Ethereum-älysopimuksilla siten, että ne toimivat yksinkertaisina tietovarastoina, joiden käyttöoikeudet on rajattu tarkasti. Ethereum tosin on julkinen lohkoketju ja käyttöoikeuksien rajaaminen ei estä tiedon lukemista suoraan lohkoista. Tätä varten on olemassa luvanvaraiset lohkoketjut, joissa tieto voidaan piilottaa täydellisesti.

Alla oleva kaavio (KUVIO 2) kuvastaa prototyypin rakennetta. Voidaan huomata, että se koostuu viidestä tietokannasta, palvelusetelirajapinnasta (Voucher API) sekä asiakasohjelmasta. Tietokannat keskustelevat keskenään rakentaakseen kokonaiskuvan käyttäjän oikeuksista ja tiedoista. Katkoviivat ovat tietokantojen välistä kommunikointia, ja yhtenäiset viivat suoria käyttäjän tai rajapinnan kutsuja. Tietokannat eivät anna lukea itseään suoraan vaan ne paljastavat itsestään lukurajapinnan (engl. Read API). Tietokannan käyttöoikeudet voidaan määrätä kirjoittamiselle ja lukemiselle erikseen (ks. KUVIO 3).



KUVIO 2. Prototyypin rakenne

```
//TownDB -> PersonDB API READ
 await databases.personDb.setReadAccess(databases.townDb.address, true);
 //TownDB -> RelationshipDB WRITE
 await databases.relationshipDb.setWriteAccess(databases.townDb.address, true);
 //CompanyDB -> PersonDB API READ
 await databases.personDb.setReadAccess(databases.companyDb.address, true);
 //CompanyDB -> RelationshipDB WRITE
 await databases.relationshipDb.setWriteAccess(databases.companyDb.address, true);
```

KUVIO 3. Käyttöoikeuksien hallinta

Tunnistautuminen

Käyttäjän henkilöllisyyden varmentaminen toteutuksessa käyttää hyödykseen Ethereumin soveltamia yksityinen-julkinen avainpareja (private-public key pair). Lähettäessään herätteen lohkoketjuun käyttäjä allekirjoittaa herätteen omalla yksityisellä avaimellaan, jolloin lohkoketju voi todentaa lähettäjän henkilöllisyyden. Älysopimuksen saadessa herätteen käyttäjältä, lähettäjän osoitteen saa selville muuttujasta msg.sender. Osoitetta verrataan tunnettuihin osoitteisiin, ja jos osoite löydetään tietokannasta, käyttäjä on tunnettu.

Prototyypissä osoitetietokanta (AddressDB) säilyttää osoitteet tunnistautumista varten (KUVIO 4).

```
function addLink(address a, string id) public write mustNotExist(a){
    linked[a] = id;
}

function getLink(address a) public view read mustExist(a) returns(string) {
    return linked[a];
}
```

KUVIO 4. Henkilöllisyyden sitominen osoitteeseen ja sen lukeminen

Kun henkilö on sidottu osoitteeseen, hänet voidaan tunnistaa nopeasti:

```
string memory link = addressDb.getLink(msg.sender);
```

KUVIO 5. Henkilön tunnistaminen

Oikeuksien määrittäminen

Tunnistamisen lisäksi on tärkeätä määrittää käyttäjän suhteet ja oikeudet. Nämä on määrätty suhdetietokannassa (RelationshipDB) (KUVIO 6). Se määrittää eri osapuolien väliset suhteet, ja näitä suhteita käytetään hyväksi oikeuksien määrittämisessä. Suhdetietokantaan voidaan tallentaa esimerkiksi työsuhde henkilön ja yrityksen välille:

```
  function addRelationship(DatabaseObject obj1, DatabaseObject obj2, string relation) public
write {
      Relationship r = new Relationship(obj1, obj2, relation, this);

      relations[obj1][obj2] = r;
      relations[obj2][obj1] = r;

      usedIds[obj1].push(obj2);
      usedIds[obj2].push(obj1);

   }
```

KUVIO 6. DatabaseObject voi olla mm. henkilö tai yritys

Esimerkiksi työsuhteen lisääminen onnistuu näin:

```
relationshipDb.addRelationship(person, company, "employed");
```

KUVIO 7. Työsuhteen lisääminen

Jos jokin toiminto vaatii, että henkilö on tietyn yrityksen työntekijä (ja työntekijä on vain yhdessä yrityksessä), voidaan tarkistaa, onko henkilö työsuhteessa (KUVIO 8).

```
  function getEmployment(string id) public view mustExist(id) read returns (DatabaseObject) {
       Person p = persons[id]; //Hae henkilöllisyys
       address[] memory relations = relationshipDb.getAllRelations(p); //Kaikki henkilön
suhteet
       //Etsitään suhde, jonka tyyppi on "employed"
       for(uint i = 0; i < relations.length; i++) {
           DatabaseObject obj = DatabaseObject (relations[i]);
           string memory r = relationshipDb.getRelation(p, obj);
           //Merkkijonojen vertaaminen Solidityllä on melko hankalaa
           if(keccak256(abi.encodePacked(r)) == keccak256(abi.encodePacked("employed")))
           {
              return obj;
           }
       }
       //Jos työsuhdetta ei löydy, keskeytetään
       require(false);
   }
```

KUVIO 8. Työsuhteen hakeminen

Nyt henkilö voidaan tunnistaa ja hänen oikeutensa määritellä, ja voidaan aloittaa uuden palvelusetelin luominen (KUVIO 9).

```
function redeemVoucher(Voucher v) public {
       string memory id = addressDb.getLink(msg.sender); //Haetaan henkilöllisyys
       DatabaseObject obj = personDb.getEmployment(id); //Haetaan työsuhde
       require(keccak256(abi.encodePacked(obj.getType())) ==
keccak256(abi.encodePacked("Company"))); //Suhde pitää olla yritykseen
       Company c = Company(obj);
       require(c == v.getProvider()); //Työnantajan täytyy olla palvelunantaja
       balances[c] += v.redeem(c);
   }
```

KUVIO 9. Palveluntarjoajan työntekijä voi lunastaa palvelusetelin

Palvelusetelin luominen

Palvelusetelin luominen vaatii, että lähettäjän henkilöllisyys ja oikeudet varmennetaan. Palvelusetelin luojan täytyy olla työsuhteessa jollekin kaupungille, jotta hän voi luoda palvelusetelin. Esimerkissä ei ole väliä, mistä potilas on. Jos oikeudet ovat kunnossa, älysopimus luo uuden palvelusetelin, jonka lääkäri allekirjoittaa. Siihen merkitään heti henkilö, jota sopimus koskee, palvelusetelin myöntänyt kaupunki ja palvelusetelin myöntänyt lääkäri sekä palvelusetelin arvo. Sen jälkeen sopimus lisätään tietokantaan, josta se on osapuolten katseltavissa. Palvelusetelin luominen kuvataan kuviossa 10.

```
function createVoucher(string socialId, uint value) public returns (string) {

        Person serviceUser = personDb.get(socialId);

        //Tarkista henkilöllisyys
        string memory doctorId = addressDb.getLink(msg.sender);
        Person doctor = personDb.get(doctorId);

        //Vaadi, että työnantaja on kaupunki
        DatabaseObject obj = personDb.getEmployment(doctorId);
        require(keccak256(abi.encodePacked(obj.getType())) ==
keccak256(abi.encodePacked("Town")));
        Town t = Town(obj);

        //Luo uusi palveluseteli
        Voucher v = new Voucher(this, serviceUser, doctor, t, value);

        vouchers[serviceUser].push(v);
        vouchers[t].push(v);

    }
```

KUVIO 10. Uuden palvelusetelin luominen

Kun henkilölle myönnetään palveluseteli, hän voi tarkastella omia seteleitään rajapinnan avulla:

```
function getVouchers() public view returns (Voucher[]){
        string memory id = addressDb.getLink(msg.sender);
        Person p = personDb.get(id);
        return vouchers[address(p)];
    }
```

KUVIO 11. Omien palveluseteleiden hakeminen

Seuraavaksi henkilön tulee valita palveluntarjoaja, joka suorittaa hänelle määrätyn hoidon. Rajapinnan kautta soveltuvat palveluntarjoajat saadaan seuraavasti:

```
  function getQualifiedCompanies(Voucher v) public view returns (Company[]) {
        Town t = v.getTown(); //Haetaan setelin myöntämiskunta
        Company[] memory companies = t.getQualifiedCompanies(); //Haetaan kaupungin valtuuttamat
yritykset

        return companies; //Palautetaan valtuutetut yritykset
    }
```

KUVIO 12. Valtuutettujen yritysten hakeminen rajapinnasta

Kun palveluntarjoaja on päätetty, voidaan se valita näin:

```
  function selectProvider(Voucher v, Company c) public {
       Person p = addressDb.getLinkedPerson(msg.sender); //Haetaan henkilö
       v.selectProvider(p,c);
   }
```

KUVIO 13. Palveluntarjoajan valitseminen rajapinnan kautta

Hoitotoimenpiteen suorittamisen jälkeen palveluntarjoaja lunastaa palvelusetelin arvon ja arvo siirtyy palveluntarjoajan tilille:

```
  function redeemVoucher(Voucher v) public {
       Person p = addressDb.getLinkedPerson(msg.sender); //Tarkastetaan henkilöllisyys
       Company company = Company(relationshipDb.getSuitableRelationships(p, "Company",
"employed")[0]); //Haetaan yritys, johon henkilö on työsuhteessa
       balances[company] += v.redeem(company); //Lisätään varat yrityksen tilille, jos yritys
on setelin palveluntarjoaja
   }
```

KUVIO 14. Palvelusetelin arvon lunastaminen

5.2.2 Tulokset

Simulaatio sivuuttaa monia reaalimaailman ongelmia ja on yksinkertainen esimerkki siitä, mitä palveluseteli lohkoketjun avulla voisi olla. On selvää, että mikäli tällainen palvelu tuotettaisiin, sitä ei toteutettaisi julkisena lohkoketjuna, missä kuka tahansa pääsisi tutkimaan lohkoja, näkemään kaiken sinne laitetun informaation ja toimimaan täysivaltaisena solmuna. Informaation näkemisen lisäksi julkiset lohkoketjut skaalautuvat huonosti eivätkä pysty tallentamaan suurta määrää tietoa, ja niiden konsensus-mekanismit kuluttavat energiaa ison määrän. Yksityisessä ja konsortiolohkoketjussa luottamus on oletettua, eikä kaikkien solmujen tarvitse osallistua siirron todentamiseen. Tämä johtaa matalampiin ylläpitokustannuksiin. Simulaatiossa toteutettujen älysopimusten periaatteet siirtyvät suoraan luvanvaraisiin lohkoketjuihin, kuten Hyperledger Fabriciin. Näissä lohkoketjuissa oikeuksien ja suhteiden määrittäminen on sisäänrakennettua ja varsinaisen tuotantomallin toteuttaminen on huomattavasti helpompaa.

Lohkoketjut rekisterinä on lakiteknisesti hankala kysymys. Toukokuussa 2018 voimaan tullut Euroopan unionin GDPR-asetus vaatii, että henkilöllä on oikeus saada tietonsa poistettua rekisteristä. Tämä on lohkoketjujen toimintaperiaatteen vastaista, eli nykylainsäädännön mukaan mitään henkilötietoja ei saa tallentaa lohkoketjuun. Lohkoketjuihin voidaan kuitenkin integroida perinteisiä järjestelmiä, joita voidaan myös käyttää älysopimuksin. Tällöin kaikki yksilöivä data voidaan jättää lohkoketjun ulkopuolelle.

On myös hankala keksiä tallennusmedia, mihin avaimen saisi tallennettua luotettavasti siten, että sitä olisi helppo käyttää. Nykyään helppo tallennusmedia voisi olla Kela-kortti, johon koodattaisiin henkilön avain, mutta kortinlukijoita ei ole yleisesti käytössä. Kenties tulevaisuudessa kaikki kansalaiset voitaisiin siruttaa, eikä erillisiä avaimia enää tarvittaisi.

6 Yhteenveto

Lohkoketjuteknologia mahdollistaa useiden toisiinsa luottamattomien osapuolten luoda ja ylläpitää yhteistä tilikirjaa tai tietokantaa ilman luotettua kolmatta osapuolta. Tämä mahdollistaa suurempaa automaatiota, kun kaikki osapuolet näkevät kaiken datan jatkuvasti. Lisäksi se mahdollistaa pienempiä kuluja, koska luotettu kolmas osapuoli voidaan jättää pois ja osapuolet voivat keskenään ylläpitää lohkoketjua omilla tietokoneillaan. Sosiaali- ja terveysalalla on lukuisia eri toimijoita ja lohkoketju mahdollistaisi vaivattomamman yhteistyön näiden toimijoiden välillä.

Esitetty palveluseteli-prototyyppi mahdollistaisi automaattisen rahaliikenteen kaikkien toimijoiden välillä sosiaali- ja terveysalalla. Prototyyppi mahdollistaa, että asiakas voisi itse valita, miltä palveluntarjoajalta hän haluaa palvelunsa hankkia ja asiakkaan saatua palvelunsa palveluntarjoaja saisi automaattisesti korvauksen palvelunmaksajalta. Tämän kaltainen ratkaisu mahdollistaisi asiakkaalle vapaamman valinnan palveluntarjoajien välillä, sekä huomattavasti suoraviivaisemman toimintamallin, jossa asiakkaan ei tarvitse miettiä korvauksia. Lisäksi ratkaisu vähentäisi kuluja, kun toiminta olisi automatisoitua toisin kuin nykyään.

Teknisesti ratkaisu on hyvin suoraviivainen, mutta lakiteknisesti lohkoketjut ovat haastavia. Puhdas lohkoketjupohjainen ratkaisu ei ole teknisesti eikä GDPR:n puitteissa mahdollinen, mutta integroimalla nykyiset järjestelmät lohkoketjuun saataisiin paras mahdollinen kompromissi.

Kaikesta huolimatta lohkoketju kuitenkin tarjoaa huomattavia hyötyjä perinteisiin järjestelmiin verrattuna, kun useat eri toimijat joutuvat työskentelemään yhdessä. Näiden hyötyjen takia asiaa tulisikin tutkia lisää ja selvittää, olisiko niiden käyttö esimerkiksi palveluseteli-prototyypin kaltaisessa toiminnassa tulevaisuudessa mahdollista ja minkälaisia lakiteknisiä asioita käyttöönotto vaatisi.

Lähteet

Anythingcrypto. 2018. How Bitcoin Mining/Block Rewards Work. Saatavilla: 20.7.2018 https://www.anythingcrypto.com/guides/bitcoin-mining-block-rewards-2018

Blockgeeks. 2018. Smart Contracts: The Blockchain Technology That Will Replace Lawyers. Blockgeeksin internetsivusto. Saatavilla: 20.7.2018 https://blockgeeks.com/guides/smart-contracts/

Enisa. 2017. Distributed Ledger Technology & Cybersecurity - Improving information security in the financial sector. The European Union Agency for Network and Information Security -raportti. Saatavilla: https://www.enisa.europa.eu/publications/blockchain-security

Greenspan, G. 2015. MultiChain Private Blockchain — White Paper. Coin Sciences Ltd. Saatavilla: 20.7.2018 https://www.multichain.com/download/MultiChain-White-Paper.pdf

Kinnunen, T., Leviäkangas, P., Kostiainen, J., Nykänen, L., Rouhiainen, K. & Finlow-Bates, K. 2017. Lohkoketjuteknologian soveltaminen ja vaikutukset liikenteessä ja viestinnässä. Liikenne- ja viestintäministeriön julkaisuja 12/2017. Saatavilla: 20.7.2018 https://julkaisut.valtioneuvosto.fi/bitstream/handle/10024/80667/LVM_12_2017_Lohkoketjuteknologian%20soveltaminen.pdf?sequence=1&isAllowed=y

Kotilainen, S. 2017. Blockchain mullistaa maailman kuin internet. Alma Media Oyj:n internetsivusto. Saatavilla: 25.7.2018 https://www.tivi.fi/Kaikki_uutiset/blockchain-mullistaa-maailman-kuin-internet-6623590

Lansiti, M. & Lakhani, K. 2017. The truth about blockchain. Harward Business Review. Saatavilla: 10.11.2017 https://hbr.org/2017/01/the-truth-about-blockchain

Lewis, A. 2015. A gentle introduction to blockchain technology. Saatavilla: 20.7.2018: https://bitsonblocks.net/2015/09/09/a-gentle-introduction-to-blockchain-technology/

Lewis, A. 2016. A gentle introduction to immutability of blockchains. Saatavilla: 20.7.2018 https://bitsonblocks.net/2016/02/29/a-gentle-introduction-to-immutability-of-blockchains/

Mulders, M. 2018. Comparison of Smart Contract Platforms. Saatavilla: https://hackernoon.com/comparison-of-smart-contract-platforms-2796e34673b7

Nofer, M., Gomber, P., Hinz, O. & Schiereck, D. 2017. Blockchain. *Blockchain. Business & Information Systems Engineering. 59(3),* 183 - 187.

Salonen, J., Halunen, K., Korhonen, H., Lähteenmäki, J., Pussinen, P., Vallivaara, V. & Ylén, P. 2018. Lohkoketjuteknologian mahdollisuudet ja hyödyt sosiaali-ja terveydenhuollossa. Valtioneuvoston selvitys- ja tutkimustoiminnan julkaisusarja 80/2017. Saatavilla: 12.12.2018 http://julkaisut.valtioneuvosto.fi/bitstream/handle/10024/160464/Lohkoketjuteknologian%20mahdollisuudet%20ja%20hyodyt%20sosiaali-%20ja%20terveydenhuollossa.pdf?sequence=1&isAllowed=y

Storås, N. 2016. Lohkoketjuteknologia pähkinäkuoressa – tämä kannattaa tietää. Alma Media Oyj: n internetsivusto. Saatavilla: 12.12.2017 https://www.tivi.fi/Kaikki_uutiset/lohkoketjuteknologia-pahkinakuoressa-tama-kannattaa-tietaa-6537904

Thompson, C. The difference between a Private, Public & Consortium Blockchain -A Simple Explanation for Dummies. Blockchain Daily News:n internetsivusto. Saatavilla: 20.7.2018 https://www.blockchaindailynews.com/The-difference-between-a-Private-Public-Consortium-Blockchain_a24681.html

Tian, F. 2018. An information System for Food Safety Monitoring in Supply Chains based on HACCP, Blockchain and Internet of Things. Doctoral thesis. WU Vienna University of Economics and Business.

Walport, M. 2016. Distributed Ledger Technology: beyond block chain. A report by the UK Government Chief Scientific Adviser. Government Office for Science.

Wüst, K. & Gervais, A. 2017. Do you need a Blockchain?. *Crypto Valley Conference on Blockchain Technology (CVCBT),* 1 - 10.

Zheng, Z., Xie, S., Dai, H., Chen, X. & Wang, H. 2017. An overview of blockchain technology: Architecture, consensus, and future trends. *IEEE International Congress on Big Data (BigData Congress) ,* 557 - 564

Informaatioteknologian tiedekunnan julkaisuja
No. 65/2018

ISBN 978-951-39-7645-3 (verkkoj.)
ISSN 2323-5004

JYVÄSKYLÄN YLIOPISTO