



Osaamista
ja oivallusta
tulevaisuuden
tekemiseen

Jari Ruuska

Tekoälyn ja koneoppimisen soveltami- nen aeroponisessa perunanviljelyssä

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Automaatiotekniikka

Insinöörityö

11.5.2020

Tekijä Otsikko	Jari Ruuska Tekoälyn ja koneoppimisen soveltaminen aeroponisessa perunanviljelyssä
Sivumäärä Aika	52 sivua + 1 liite 11.5.2020
Tutkinto	insinööri (AMK)
Tutkinto-ohjelma	sähkö- ja automaatiotekniikan koulutusohjelma
Ammatillinen pääaine	automaatiotekniikka
Ohjaaja	tutkijaopettaja Aarne Klemetti
<p>Insinööriyön aiheeksi valittiin tekoälyn ja koneoppimisen soveltaminen aeroponisessa perunanviljelyssä. Työn tavoitteeksi määriteltiin koneoppimisalgoritmin muodostaminen, jolla parantaa automatisoitua aeroponista perunankasvatusprosessia. Työn tavoitteen ohella työn aikana toteutettiin testialustan automatisointi. Testialustan antamaa dataa käytettiin koneoppimisalgoritmin toteutukseen.</p> <p>Työ aloitettiin kartoittamalla ja hankkimalla testialustan automatisointiin tarvittavat välineet, kuten SIMATIC ET 200SP PLC-logiikka, lämpötilaa ja hiilidioksiditasoa mittaavat anturit sekä MindConnect-laite. MindConnect-laitteen avulla siirrettiin dataa MindSphere-IoT-raja-pintasovellukseen. Testialustan kytkennät suoritettiin, jonka jälkeen PLC-logiikka ohjelmointiin TIA Portal -ohjelmistolla. Ohjelmointikielenä käytettiin FBD-kieltä. MindSphere-sovellus otettiin käyttöön ja PLC-logiikan välittämä data siirrettiin sovellukseen. Anturien tuottama data tuotiin .csv-tiedostomuodossa aikajaksoisena tiedostona sovelluksesta, jonka jälkeen dataa siivottiin käyttökelpoiseen kuntoon.</p> <p>Koneoppimisalgoritmin muodostukseen käytettiin Python-ohjelmointikieltä sekä Anaconda-jakelualustaa. Algoritmin mallina käytettiin lineaariregressiota, jonka muodostukseen käytettiin testialustan tuottamaa dataa. Malli todettiin toimivaksi mallissa muodostetun lineaarisen suoran asettumisen ja algoritmin antaman vastauksen perusteella. Tuloksista pääteltiin lineaariregressiomallin sopivan pitkäaikaiselle datankeräykselle. Aeroponisessa perunankasvatuksessa mallin käyttö todettiin sopivan ohjekäyttöön automatisoidun sadonkasvatuksen rinnalle.</p>	
Avainsanat	aeroponinen viljely, tekoäly, koneoppiminen, automatisointi, MindSphere, algoritmi

Author Title	Jari Ruuska Application of Artificial Intelligence and Machine Learning in the Aeroponic Cultivation of potato
Number of Pages Date	52 pages + 1 appendix 11 May 2020
Degree	Bachelor of Engineering
Degree Programme	Electrical and Automation Engineering
Professional Major	Automation Technology
Instructor	Aarne Klemetti, Researching Lecturer
<p>The target of the thesis project was to write a Machine Learning algorithm to upgrade the aeroponic potato growth process. Along the process, the automation of a test setting was carried out. The data of the test setting was used as a part of writing the algorithm.</p> <p>The work was started by mapping and acquiring equipment, which was needed for automation of the test setting. Equipment were SIMATIC ET 200SP PLC controller, sensors for temperature and carbon dioxide measurement and a MindConnect device. The MindConnect device was used for a MindSphere connection, which is an IoT interface application. The acquired equipment were linked together. The PLC controller was programmed by FBD programming language. The programming was performed by TIA Portal program. The MindSphere application was brought into use. The data conveyed by PLC controller was transferred to the MindSphere application. The data produced by sensors was imported from the MindSphere application as a time series .csv file. The file was tidied up for better usage.</p> <p>The Python programming language and Anaconda distribution were used for constructing the Machine Learning algorithm. The result of the work is a linear regression model. Functionality of the model was confirmed by feedback of the model. The linear regression model was decided to be suitable for long-time data collecting. The model was found to be useful instructional addition to automated cultivation.</p>	
Keywords	aeroponics, artificial intelligence, machine learning, automation, MindSphere, algorithm

Sisällys

1	Johdanto	1
2	Aeroponinen perunanviljely	2
2.1	Perunankasvatuksen lähtötilanne Urban Farm Labissa	3
2.2	Automatisoinnin syyt	4
2.3	Automatisoinnin tietopohja	5
3	Tekoäly	6
3.1	Tekoälyn historia	6
3.2	Tekoäly nykyään	7
4	Koneoppiminen	8
4.1	Lukujen käsittely	9
4.2	Tehtävät	10
4.2.1	Ohjattu oppiminen	11
4.2.2	Ohjaamaton oppiminen	14
4.2.3	Vahvistusoppiminen	15
4.3	Mallit	16
4.3.1	Geometrinen malli	16
4.3.2	Todennäköisyysmalli	17
4.3.3	Looginen malli	17
5	Projektissa käytetyt välineet ja ohjelmat sekä työn kulku	18
5.1	Anturit	19
5.1.1	Pt100-anturi ja sen tarvitsema lähetin	19
5.1.2	EE800 CO ₂ - ja lämpötilalähetin	20
5.1.3	Kapasitiivinen lähestymisanturi E2K-F10MC1	22
5.2	PLC-logiikka ja sen käyttämät moduulit	23
5.2.1	SIMATIC ET 200SP - CPU 1512SP-1 PN	23
5.2.2	Server-moduuli ja BaseUnit-moduulit	24
5.2.3	Analoginen ja digitaalinen lisämoduuli	25
5.3	TIA Portal -ohjelmisto	25

5.3.1	SIMATIC-käsite ja TIA Portal -ohjelmiston sisältö	26
5.3.2	Function Block Diagram -ohjelmointikieli	26
5.4	MindSphere	27
5.5	Koneoppimiseen käytettävät menetelmät	29
5.5.1	Python	29
5.5.2	TensorFlow	29
5.5.3	Anaconda	30
5.6	Työosuuden kulku	31
5.6.1	Anturien ja PLC-logiikan kartoitus	31
5.6.2	Projektissa suoritettu asennus asennuskoteloon	32
5.6.3	PLC-ohjelman luonti	34
5.6.4	Verkkojen luonti	35
5.6.5	Yhteyden luonti MindSphere-sovellukseen ja datojen siirto	36
5.6.6	Datan tuonti MindSphere-sovelluksesta ja sen siistiminen	38
5.6.7	Koneoppimissovelluksen luominen	39
6	Tulokset ja analysointi	40
6.1	Tulokset	40
6.2	Analyysi	43
7	Yhteenveto	45
	Lähteet	47
	Liitteet	
	Liite 1. MindSphere-sovelluksen käyttöönotto-ohjeet	

1 Johdanto

Opinnäytetyöraportti kuvaa projektia, jonka tarkoitus on parantaa automatisoitua aeroponisen perunankasvatuksen prosessia tekoälyn ja koneoppimisen keinoin. Aeroponinen perunankasvatus on maaperätön perunoiden kasvatustapa, jossa perunat roikkuvat ilmassa tavallisesti säiliön kaltaisessa kasvatustavassa. Kasvatustavassa perunoiden juurille suihkutetaan ravintoaineliuosta. Projektissa selvitettiin aeroponisen kasvatusprosessin automatisointia, koska sen avulla on mahdollista vähentää prosessissa ihmistyövoimalla tehtävää tarkkailua. Projektissa luotiin aeroponisen kasvatussäiliön automatisoinnin testialusta, jolla oli mahdollista kerätä dataa ympäristön lämpötilasta ja hiilidioksiditasosta.

Tekoälytieteen osajoukon koneoppimisen menetelmiä käytettiin koneoppimisalgoritmin luontiin. Algoritmilla tarkoitetaan ongelmanratkaisevaa keinoa, kuten esimerkiksi tietokoneella tehtävää ohjelmanpätkeä. Testialustan keräämä data voitiin siirtää projektin aikana käytössä olleeseen IoT-rajapintasovellukseen, johon dataa kerättiin aikajaksoissa muodossa. Aikajaksoista dataa käytettiin koneoppimisalgoritmin luomiseen. IoT-rajapintasovelluksella tarkoitetaan tässä projektissa Siemensin MindSphere-sovellusta, joka on pilvipohjainen selaimessa toimiva ympäristö oman datan siirtoon ja tarkasteluun. Tekoäly- ja koneoppiminen-sanat ovat nykypäivänä suosittuja termejä, ja niiden on nähty kulkevan samoissa lauseissa, vaikka eroavatkin toisistaan merkityksiltään. Termit kuvastavat tieteenaloja, joiden käytäntöjen avulla luodaan ihmisten tekemien töiden helpottamiseksi käytettäviä ohjelmia. Tekoäly on yleistermi tällaisille ohjelmille, kun taas koneoppiminen on tekoälyn osajoukko. Koneoppimisen avulla voidaan luoda tietokonealgoritmeja, jotka kykenevät kokemuksensa perusteella parantamaan antamaansa vastausta.

Opinnäytetyöprojekti suoritettiin osana Digi-Salama -hanketta, joka on Metropolia Ammattikorkeakoulun ja Vantaan kaupungin toteuttama hanke yhteistyössä Suomen Automaatioseuran ry:n, Suomen Robotiikkayhdistys ry:n, Tampereen yliopiston hallinnoiman TRINITY-hankkeen ja VTT:n hallinnoiman DIH2 -hankkeen kanssa. Digi-Salama -hankkeen tavoitteena on tarjota pk-yrityksille kilpailukykyä ja uusia ratkaisuja Digi-Salama-

projekteissa syntyvien tuotosten kautta. Digi-Salama -hankkeen projektit koostuvat viidestä digitaalisen Industry 4.0 -käsitteen teemasta, jotka ovat cobotiikka, autonomiset liikkuvat robotit, Digital Twin -teknologia, tekoäly ja koneoppiminen automaatioissa sekä VR/AR/Holodeck. Tässä projektissa tutustuttiin teemoista tekoälyyn ja koneoppimiseen automaatioissa. [1.]

Projektin teeman tutkittavaksi käyttökohteeksi tuli Metropolia Ammattikorkeakoulun Myyrmäen toimipisteen tiloissa toimivan Urban Farm Lab -testilaboration aeroponisen perunankasvatuksen prosessi. Urban Farm Lab on Metropolia Ammattikorkeakoulun, Sitran ja viiden hankkeeseen kuuluvan yrityksen urbaanin ruoantuotannon testilaboratoriotila. Tilassa tuotetaan ja kokeillaan erilaisia kaupunkiympäristöissä mahdollistettavia ruoantuotantoteknologioita, kuten perunankasvatusta aeroponisella menetelmällä. [2.]

Opinnäytetyöraportissa tutustutaan aluksi aeroponiseen perunankasvatukseen yleisesti, sen tilaan Urban Farm Lab -tilassa sekä kasvattajan ja automatisoijan tarvitsemiin tietoihin kasvatuksesta. Tämän jälkeen tutustutaan yleisesti tekoälyyn ja koneoppimiseen. Tekoälystä esitellään sen historiaa ja käyttökohteita. Koneoppimiseen tutustutaan algoritmien luonnin päätekijöihin ja muutamiin esimerkkeihin, jotka antavat kokonaiskuvaa tieteenalasta. Luvussa 5 käydään läpi projektissa käytettäviä välineitä ja tietokoneohjelmia sekä käydään läpi projektin kulku. Luvussa 6 esitellään projektin tulokset sekä analyysi. Luvussa 7 on loppuyhteenveto.

2 Aeroponinen perunanviljely

Aeroponinen viljely on kasvien kasvatustapa, jossa kasvin juuret riippuvat avoimessa tai suljetussa säiliössä. Säiliön sisällä oleviin juuriin suihkutetaan jaksoittaisesti tai jatkuvasti ravintopitoista liuosta. Tällä kasvatustavalla kasvi saa runsaasti happea, vettä ja ravintoa, jotka mahdollistavat kasvin täysikasvuisuuden. [3, s.1.]

Lyhyesti ilmaistuna kasvit kasvavat maaperättömässä ympäristössä pienellä määrällä vettä. Tekniikka on kehitetty 1920-luvulla kasvitieteilijöiden toimesta, jotka tutkivat kasvien juurirakennetta. Tekniikka ei kehittynyt akateemista tutkimusta pidemmälle ennen vuosikymmentä 1970, jolloin kiinnostuttiin hydroponisesta viljelystä. Hydroponinen viljely

on samanlainen kasvatustapa kuin aeroponinen viljely, mutta kasvien juuret kasvatetaan hydroponisessa kasvatustavassa ravintopitoisessa vedessä. Aeroponinen viljely yleistyi viimeistään NASA:n kiinnostuessa tekniikasta viraston testatessa kasvatustapaa avaruuslentoja varten. [4.]

2.1 Perunankasvatuksen lähtötilanne Urban Farm Labissa

Metropolian Myyrmäen toimipisteessä toimivassa Urban Farm Lab -testilaboratoriotilassa kasvatetaan aeroponisella kasvatustavalla perunoita Siukkula Oy -nimisen yrityksen toimesta. Yrityksen toimitusjohtaja Raine Hermans kertoo Vantaan Sanomien artikkelissa, että yrityksen tavoitteena on toteuttaa viljelylaitteisto, jonka avulla perunoita voidaan kasvattaa esimerkiksi kotioloissa kerrostaloasunnossa. Hermans kertoo artikkelissa, että yritys on saanut kasvatettua aeroponisella kasvatustavalla 40 perunaa yhtä vartta kohden, mitä hän vertaa peltoviljelyllä saatuun 4–6 perunaan yhtä vartta kohden. Hän kertoo myös, että aeroponisessa viljelyssä samaa vartta voidaan käyttää uudelleen sadon keräämisen jälkeen. [5.]

Liikesalaisuuksista johtuen Metropolian Siukkula Oy:n aeroponisen perunankasvatussäiliön tarkkaa toimintaa ei ole tiedotettu yrityksen ulkopuolelle, mutta havaintojen perusteella säiliössä olevaa kasvustoa syötetään tasaisin väliajoin pumpulla, joka suihkuttaa juurille ravintopitoista liuosta. Säiliön sisällä olevat juuret roikkuvat täydellisessä pimeydessä. Tosin säiliössä on tarvittaessa avattava luukku sadon tarkastelua ja keruuta sekä säiliön puhdistusta varten. Perunan varret roikkuvat niille tehdyissä säiliön läpivienneissä. Säiliön ulkopuolella kasvavat perunan versot kylpevät niille keinotekoisesti syötetyssä valossa. Viljelyjärjestelmää on automatisoitu tasaisen ravinnonsyötön lisäksi kiinteistöautomaatiojärjestelmällä ja anturoinnilla. Anturointiin kuuluvat

- kolme lämpötila-anturia, jotka mittaavat perunan versojen ja juurien sekä säiliöön suihkutettavan veden lämpötilaa
- hiilidioksidianturi, joka on sijoitettu säiliön kannen päälle mittaamaan kasvin versojen hiilidioksiditasoa
- kosteusanturi, joka mittaa perunan kasvuympäristön kosteustasoa

- pinnankorkeuden tietyn tason ylityksestä aktivoituva anturi.

Anturien syöttämää dataa voidaan tarkkailla rakennusautomaatioratkaisuja tuottavan Oumanin valmistaman ohjelmoitavan automaatioyksikön Ouflex A:n näyttöpaneelistä. Kuvassa 1 on Urban Farm Lab -tilan aeroponinen perunankasvatussäiliö.



Kuva 1. Aeroponinen perunankasvatussäiliö, josta kasvavat perunanversot kylpevät keinotekoisessa valossa. Säiliön sisäpuolella kasvavat perunoiden juuret. [5.]

2.2 Automatisoinnin syyt

Aeroponisen perunankasvatuksen etuihin voidaan lukea perinteiseen peltokasvatukseen verrattuna perunan ravintoaineen hallinta ja säätely sekä mahdollisten tautien ja ötoköiden puuttuminen. Edellä mainitut asiat vaikuttavat aeroponisen perunan sadon määrään ja perunan laatuun. Aeroponisessa kasvatuksessa on otettava huomioon peltoviljelyä tarkemmin vallitsevat olosuhteet, jolloin automatisointi ja käytettävissä olevat

anturit helpottavat kasvatusprosessia. Automatisoinnin avulla voidaan optimoida kasvu-
prosessia ja kasvuprosessiin sisältyviä kasvuun vaikuttavia tekijöitä sekä parantaa sa-
don onnistumisen todennäköisyyttä. Viljelyyn käytettävää prosessia suunniteltaessa
sekä prosessin aikana on otettava huomioon mahdolliset käyntivirheet, kuten vesipum-
pun tai annosteltavan ravintoainelinjan vaurioituminen sekä tihkusuuttimien tukkeutumi-
nen. Edellä mainitut syyt voivat johtaa kasvuston kuolemaan, mikäli ongelmaa ei huo-
mata ja korjata ajoissa. [6, s. 2–3.]

2.3 Automatisoinnin tietopohja

Aeroponista kasvatusprosessia automatisoitaessa anturoinnin kannalta tärkeitä tietoja
ovat kasville syötettävän ravintoliuoksen lämpötila, happamuus ja sähkönjohtavuuskyky.
Parametreille on määritelty tietty arvoväli, jonka ylittäminen tai alittaminen aiheuttavat
ongelmia kasvin kasvussa. Lisäksi kasvuston kasvua voidaan optimoida seuraavilla pa-
rametreilla:

- juurille kohdistuvan veden ja ravintoaineen sumutukseen käytettävä aika
- juurille kohdistuvan veden ja ravintoaineen sumutusvälien aika
- kasvuympäristön ilman lämpötila
- kasvuympäristön ilman suhteellinen kosteus
- kasvin saaman valon intensiteetti
- kasvuympäristön ilman hiilidioksidipitoisuus. [6, s. 5–6.]

Liuoksen sähkönjohtavuuskykyä mitataan mittaamalla liuokseen liottuneiden suolojen
määrää. Ravintoaineen happamuutta mitataan tutkimalla liuoksen pH-arvoa. Aero-
ponisessa prosessissa vettä ja ravintoainetta kierrätetään säännöllisesti sumutuksien
johdosta, minkä takia sähkönjohtokyvyn ja happamuuden säännölliset mittaukset mah-
dollistavat onnistuneen kasvun arvojen pysyessä raja-arvojen välissä. Molempien arvo-
jen mittaamisen käytetään pH- ja EC-anturia.

Prosessin lämpötila vastaa suoraan kasvun kehitystä ja tiheyttä. Aeroonisessa prosessissa ilman ja ravintoliuoksen lämpötilan hallintaa tehdään nopean kasvun mahdollistamiseksi. Lämpötilan kohotessa kemialliset prosessit etenevät nopeammalla tahdilla ja heikentävät entsyymien toimintoja. Optimaalinen lämpötilaväli kaikille kasveille on 15–25 °C. Lämpötilan mittaukseen käytetään lämpötila-anturia. [7, s.8.]

Perunan kasvuun kuluttamaansa hiilidioksidin tasoa voidaan mitata CO₂-anturilla. Tutkimusten mukaan hiilidioksiditason kohoaminen vaikuttaa sadon määrään positiivisesti kasvattamalla perunoiden kokoa [8, s.1].

3 Tekoäly

Tekoäly ja koneoppiminen käsitteet eroavat toisistaan, vaikka vaikuttavatkin samoilta. Tekoälyllä tarkoitetaan käsitettä, jossa luodaan älykkäitä koneita tai järjestelmiä, jotka osaavat simuloida ihmisen käytöstä. Koneoppiminen taas on tekoälyn osajoukko, jossa konetta opetetaan sille syötetyllä datalla ilman että sitä ohjataan tekemään tiettyjä toimintoja. [9; 10.]

Tekoälyllä tarkoitetaan laajalle levinnyttä tietojenkäsittelytieteen alaa, jossa keskitytään rakentamaan älykkäitä laitteita tai sovelluksia suorittamaan tehtäviä, jotka tavallisesti suoritetaan ihmisen toimesta [11]. Tekoälystä on syntynyt monenlaisia käsityksiä, jotka johtuvat tieteiselokuvien, markkinointi-ihmisten ja tiedemiesten luomista mielikuvista älystä, jotka pystyvät ylittämään ihmismielen kyvyn ratkaista ongelmia. Tekoälyjen ongelmanratkaisukykyä tutkiessa pitää muistaa, että tekoälyn ja ihmisen älykkyydellä ei ole yhteistä tekijää. Tekoäly toimii ihmisen luomien algoritmien välityksellä, joiden ansiosta tekoäly pystyy ratkaisemaan käsillä olevan tehtävän joko ihmisen toimintaa jäljittelevillä menetelmillä tai ilman niitä. [12, s. 1, 7, 12.]

3.1 Tekoälyn historia

Tekoälyn ensimmäiset askeleet voidaan katsoa saaneen alkunsa Bombe- ja Enigma-laitteiden käytöstä toisen maailmansodan aikaan kryptattujen viestien purku- ja luonti-

töissä. Molempien koneiden voidaan katsoa olevan lähtökohta koneoppimiselle. Nykyaikajan tekoälyn tieteellinen tutkimus sai alkusysäyksen vuonna 1956 Dartmouthin Collegien konferenssitapaamisesta, jossa konferenssiin osallistujat puhuivat tekoälyn tulevaisuudesta.

Ensimmäisinä vuosina kehitettiin algoritmeja muun muassa matematiikan ongelmien ratkaisuun ja robottien opettamiseen. 1970- ja 1980-luvuilla tekoälyn tutkimus kärsi rahoitusvaikeuksista ja tietokoneiden muistien pienikokoisuudesta, joka vaikutti paljon muistia ja prosessointikykyä vaativiin tehtäviin. 1990-luvulla Yhdysvalloissa kiinnostuttiin uudelleen tekoälyn tutkimuksesta ja 2000-luvulle tultaessa tietokoneiden kehitys auttoi tekoälyn tutkimuksen suorittamisessa. Viimeisten vuosien aikana tietotekniikan edelläkävijät, kuten Amazon ja Google, ovat käyttäneet koneoppimista kaupalliseksi edukseen ja ovat pystyneet edistämään tekoälyn tulevaisuutta muun muassa konenäköön ja puheentunnistukseen kohdistuvilla tutkimuksilla. [13.]

3.2 Tekoäly nykyään

Tekoälyjä voidaan huomata ympäristöstä monien sovelluksien käytöstä, kuten esimerkiksi asiakaspalvelurobotista, joka seuraa ennalta määrättyjä käskyjä ja vastaa käyttäjän kysymyksiin tunnistamien sanojensa perusteella, autojen nykyaikaisista jarrutusjärjestelmistä, joiden takana oleva tekoäly ottaa auton toimintoja hallintaansa vaaratilanteen synnyessä sekä luottokorttien petoksentunnistussovelluksista, jotka huomaavat luottokortteihin kohdistuvan epätavallisen käytöksen. [12, s.17–18.] Tekoälytutkimus sisältää nykyään monia osajoukkoja, kuten taulukko 1 näyttää.

Taulukko 1. Tekoälyn osajoukot [14].

Osajoukko	Lyhyesti	Sovelluksia
Hermoverkot	Verkkorakenne, joka imitoi biologisia hermoverkkoja	Kappaleentunnistus, kuvanpakkaus
Geneettisen las-kennan evoluutio	Aiemmin tuotetun kappaleen uudelleentuotantoa parannettuna versiona	Autonosien suunnittelu, robotit
Kuvantunnistus	Mallin ja/tai ohjaukskäskyn saanti visuaalisesta datasta	Armeijakäytössä esimerkiksi vihollisten tunnistus, maataloudessa optinen erottelu
Puheentunnistus	Teknologia, jossa harjoitetaan toimintoja kuten puheentunnistusta tai puhetta tekstiksi	Puheentunnistussovellukset, kännyköiden puheentunnistus
Robottiikka	Teknologia, jossa ohjelmoitua mekaaniset laitteet tekevät työn ihmisiä tehokkaammin ja tarkemmin	Search and Rescue -robotit, vedenalaisten tehtävien robotit
Asiantuntijajärjestelmät	Tekoälyyn perustuva järjestelmä, jota tarvitaan ihmistaitoja vaativiin tehtäviin	sanantarkastusohjelma, F22 Raptor -lentokoneen tutkajajärjestelmä
Luonnollisen kielien käsittely	Teknologia, jossa tietokone toimii ja vastaa käyttäjän luonnollisella kielellä	Kielentunnistus, oikeinkirjoitus
Koneoppiminen	Koneoppimisohjelmat tunnistavat kuvioita datasta ja muokkaavat ohjelman toimintaa	Anti-spam -ohjelma, kuvan luokittelu

4 Koneoppiminen

Koneoppiminen on tekoälyn osajoukko, jossa käsiteltävät algoritmit ja järjestelmät pysyvät parantamaan omaa tietoaan tai suorituskykyään kokemuksen kautta. Koneoppimista käyttämällä saatu lopputulema koostuu kolmesta päätekijästä, jotka ovat käsillä olevassa ongelmassa käytetyt luvut, lukujen perusteella valittu tehtävä ja lopuksi koneoppimista käyttäen ulostulona saatu malli. [15, s. 3.] Tätä lopputulemaa voidaan kutsua algoritmiksi. Algoritmilla eli ongelmanratkaisevalla menettelytavalla tarkoitetaan tänä päivänä tietokoneelle syötettävää sääntöjoukkoa tarkoituksenaan saavuttaa päämäärä eli ongelman ratkaiseminen [16]. Tässä luvussa tutustutaan koneoppimisessa käytettyihin tekijöihin, joiden avulla luodaan algoritmi ongelman ratkaisuun.

Algoritmeissa käytettävä data ei ole itsestään selvää vaan tarvitsee mallien luontia varten tarkastelua ja mahdollista muokkausta. Koneoppimisessa käytettävät tehtävät riippuvat suuresti syötettävästä datasta sekä datan oikeasta käyttötavasta tehtävää valitessa. Tehtävät-alaluvussa olevat tehtävät antavat esimaun koneoppimiseen tutustuvalla. Algoritmien maaleina olevat mallit riippuvat syötetystä datasta ja valituista tehtävistä.

4.1 Lukujen käsittely

Luvut koostuvat käsillä olevasta datasta. Dataa joudutaan useasti käsittelemään ennen käyttöä ja parantamaan sitä ennen tehtävän ja mallin muodostamista. Datan muokkauksen takia koneoppiminen on itseään toistava prosessi, sillä vasta mallin muokkauksen jälkeen tiedetään datan kyky mallin luomiseen. Usein mallin muodostamisen jälkeenkin dataa joudutaan muokkaamaan uudelleen mallin parantamista varten. [15, s. 48.]

Datan laatu ja datan hyödyllisen tiedon määrä määrittelevät sen, kuinka hyvin koneoppimisalgoritmi voi oppia. Käytössä olevasta datasta voi puuttua erinäisten syiden takia osa datasta, jolloin datasta voidaan joutua poistamaan tietoa tai korvaamaan puuttuva data jollakin toisella datalla. Käsiteltävä data koneoppimiskäytössä on yleensä numeerista dataa, kuten kokonaisnumeroita tai desimaalilukuja. Joissain tapauksissa dataa voi olla sen verran paljon, että käsiteltävä data sisältää sarakkeen, jossa on pelkkää sanalla viitattua dataa. Esimerkkinä voidaan käyttää paidan kokoa, jolloin sarakkeessa käytetään kirjaimia 'S', 'M' ja 'L'. Selvyiden ja koneoppimisalgoritmin kannalta sanallinen data voidaan muuttaa numeeriseen muotoon. Esimerkissä paidat voidaan luokitella numeerisesti kokonaisluvuilla 1, 2 ja 3, jolloin 'S' on 1, 'M' on 2 ja 'L' on 3. [17, kpl 4, 4.2.]

Dataa voidaan joutua käsittelemään monella tavalla ennen ja jälkeen suunnitellun koneoppimisalgoritmin muodostuksen, mutta myös sitä tehdessä. Dataa joudutaan osittelemaan eri datajoukoiksi opetusta ja testausta varten algoritmissa [17, kpl 4.3]. Opetusjoukko on osa käytettävästä datasta, joka sovittaa mallin. Opetuksen jälkeen malli käyttää tätä opetusjoukkoa vertausjoukkonaan muita mallin käytössä olevia joukkoja kohtaan. Testijoukkoa käytetään mallin lopullisena sovittajana, joka koostuu pienemmästä määrästä dataa kuin opetusjoukko. [18.] Käytännössä opetusjoukko tarjoaa suurimman

osan algoritmiin syötetystä datasta, jonka avulla voidaan sovittaa malli. Testijoukko tarjoaa lopullisen tarkistuksen mallin toiminnasta.

Dataa voidaan myös datajoukkojen muodostuksen lisäksi skaalata. Skaalausta tarvitaan siinä tapauksessa, että kaksi datasaraketta ovat skaalattu erisuuruiksi. Esimerkiksi toisen sarakkeen luvut ovat skaalattu 1:stä 10:een ja toisen sarakkeen luvut 1:stä 100 000:een. Datan skaalaukseen käytettävät termit ovat normalisointi ja standardointi. Datan normalisoinnissa luvut skaalataan 0:n ja 1:n välille. Datan standardoinnissa taas käytämme tietyn datasarakkeen keskiarvoa ja keskihajontaa määrittäessä arvoväliä lähelle nollaa. [17, kpl 4.4.] Standardoinnissa käytetty kaava löytyy kaavasta 1.

$$x_{std}^{(i)} = \frac{x^{(i)} - \mu_x}{\sigma_x} \quad (1)$$

$x_{std}^{(i)}$ on standardoitu luku kaavan syötetystä termistä x

$x^{(i)}$ on kaavaan syötetty luku datajoukosta x

μ_x on datajoukon x keskiarvo

σ_x on datajoukon x normaalijakauma

Datan käsittelyssä voidaan törmätä ylisovitukseen (*engl. overfitting*), joka tarkoittaa mallin mallintavan liian tarkasti opetusjoukkoa. Ylisovitusta tapahtuu, kun käytössä oleva malli oppii opetusjoukon antaman datan sekä siellä olevat heilahtelut, jotka tallentuvat malliin. Ongelma muodostuu sen jälkeen, kun mallille syötetyssä uudessa datassa ei heilahteluja tapahdu. Malli on yleistänyt opetusjoukon datan, mikä vaikuttaa käyttäytymiseen uuteen dataan verrattuna. [19.] Syy ylisovitukseen on se, että malli on liian monimutkainen eikä se sovi annettuun dataan. Ylisovituksen korjaukseen auttavat muun muassa opetusdatan lisäkeräys tai mallin vaihto sellaiseen, jolla on vähemmän käsiteltäviä parametreja. [17, kpl 4.5.]

4.2 Tehtävät

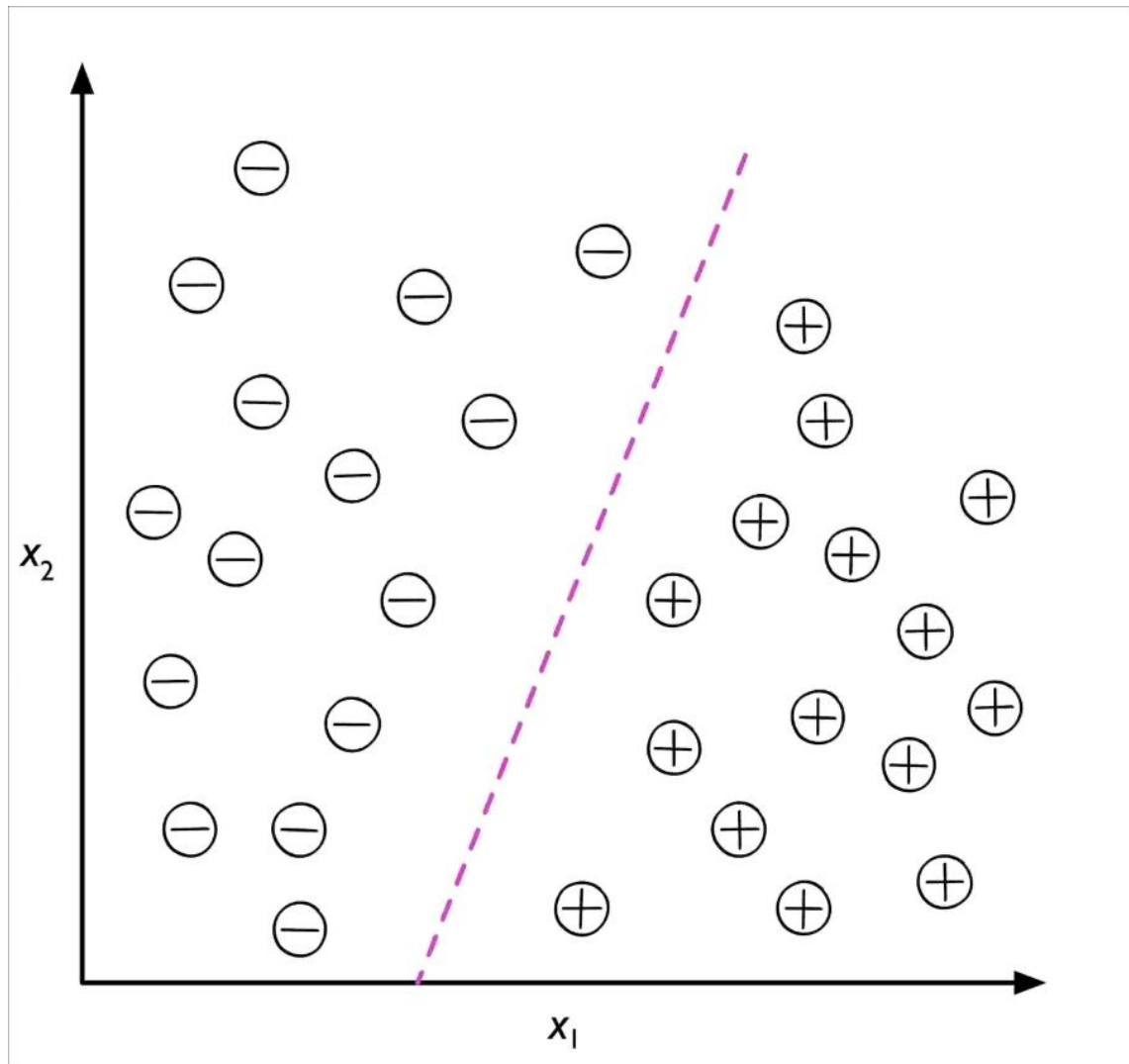
Koneoppimisalgoritmeja muodostetaan datan, mallin ja tehtävien välityksellä. Tehtävä valitaan käsillä olevan datan perusteella muodostamaan malli. Tehtävät jaetaan kolmeen tyyppiin, jotka ovat ohjattu oppiminen, ohjaamaton oppiminen ja vahvistusoppiminen.

4.2.1 Ohjattu oppiminen

Ohjatun oppimisen koneoppimistyyppin tunnusmerkit ovat nimetyn datan käyttö ja muodostetun mallin antama palaute, jonka perusteella voidaan suorittaa ennuste. Mallin muodostamisessa käytetään nimettyä dataa ja sopivaa tehtävää, joiden perusteella opetettu data antaa käyttäjälleen mallin, jonka avulla käyttäjä saa suoritettua ennustuksia syötetystä datasta. Termi ohjattu tyyppin nimen alussa tarkoittaa sitä, että malliin syötetty datajoukko on tunnettu, joten käyttäjä ikään kuin ohjailee mallia nimetyllä datajoukolla haluttuun lopputulokseen. Algoritmi suorittaa itseään uudelleen ja uudelleen tehdäkseen ennustuksia opetusdatajoukosta. Ennustusten teko loppuu algoritmin saavuttaessa maalinsa eli hyväksyttävän suoritustason. Ohjatun oppimisen tyyppi sisältää kaksi tehtäväkategoriaa, jotka ovat luokittelu sekä regressio. [17, kpl 1.2; 20.]

Luokittelulla viitataan koneoppimisessa tehtävään, jossa tapahtuu yksittäisen datatiedon luokkatunnuksen ennustaminen ja nimeäminen. Mallin muodostamisen kannalta opetusdatajoukko pitää olla tarpeeksi suuri, jotta malli pystyy oppimaan luokittelemaan dataa eri luokkiin. Opetusdatajoukon pitää olla tarpeeksi kattava ja sen pitää sisältää monia esimerkkejä jokaisesta luokkatunnuksesta. Esimerkkinä luokittelussa käytetään binäärisen luokittelun tapausta ”spam”-posteista. [21.] Luotu malli voi luokitella dataa joko ”spam”- tai ”no spam”-luokkaan riippuen käyttäjän datalle antamasta painotuksista. Kaksiluokkaisen binäärisen luokittelun lisäksi on olemassa useamman luokan luokittelutehtäviä [21].

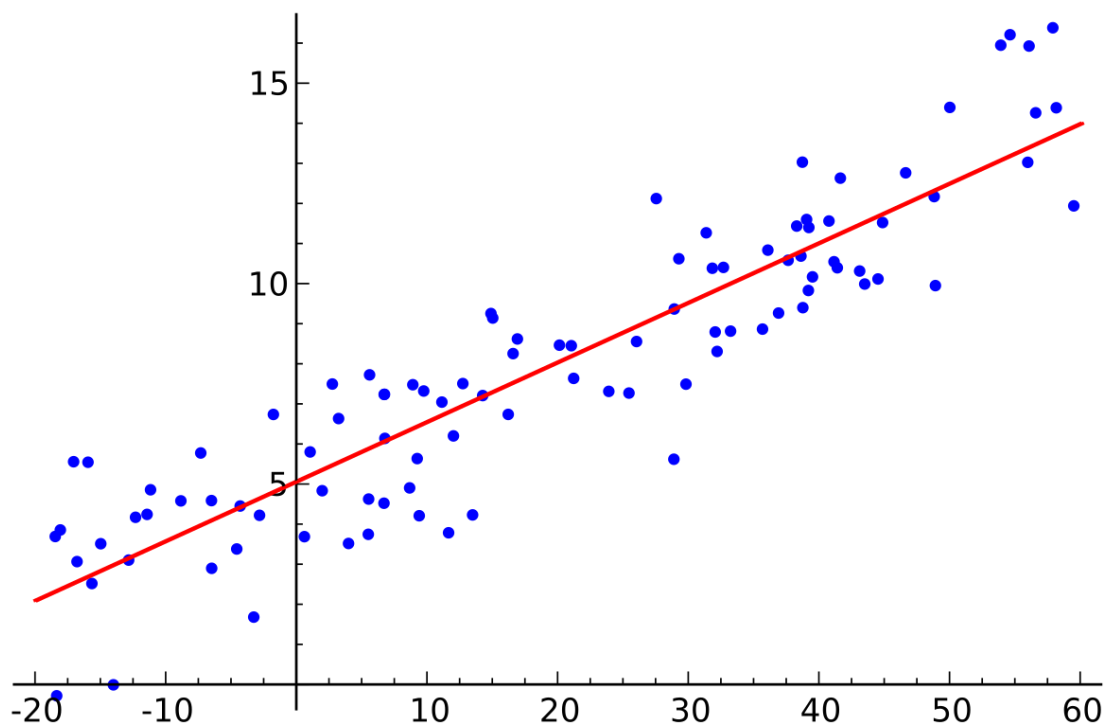
Kuvassa 2 nähdään esimerkki binäärisestä luokittelusta, jossa kuvaajassa esiintyvät datapisteet sisältävät negatiivisia ja positiivisia näytteitä. Datapisteitä määrittelevät datajoukot x_1 ja x_2 . Datapisteitä erottaa katkoviivalla merkitty rajaviiva, joiden molemmilla puolilla sijaitsevat vastakkaisen luokan pisteet. Luokittelualgoritmi erottelee uuden datan sen sisältämän tiedon perusteella riippuen datapisteen $x_{1:n}$ ja $x_{2:n}$ arvoista. [17, kpl 1.2.]



Kuva 2. Esimerkki luokittelusta [17, kpl 1.2].

Regressioanalyysin tehtävänä on kartoittaa ja arvioida funktio (f) tulomuuttujan (X) ja jatkuvan lähtömuuttujan (y) datan perusteella. Lähtömuuttuja on jokin reaaliarvoinen muuttuja, kuten kokonaisluku tai desimaaliluku. Muuttujat ovat usein määriä tai kokoja. [22.] Tavoitteena on löytää muuttujien välille yhteys, jonka avulla voidaan ennustaa regressioanalyysin lopputulos [17, kpl 1.2]. Tulomuuttujia (X) voi olla useampi määrä regressioanalyysissä. Regressioanalyysissä tehdessä voidaan törmätä käsitteisiin lineaariregressio ja logistinen regressio. Analyysien erona on se, että lineaariregressiota käytetään jatkuvan tulon ennustuksiin, kun taas logistinen regressio kuuluu luokitteluun kuuluvan diskreettien tulojen ennustukseen. [23.]

Esimerkki *lineaariregressiosta* on yksinkertainen lineaariregressio, jossa muodostettava malli koostuu yhden tulomuuttujan (X) ja yhden lähtömuuttujan (y) antaman datan läpikäymisestä erilaisten matemaattisten funktioiden avulla. Malli muuttuu sitä tarkemmaksi, mitä useampi kerta dataa käydään läpi. Kuvassa 3 on esimerkki yksinkertaisesta lineaariregressiosta. Siniset pisteet ovat muuttujien välisiä datapisteitä (X, y) ja punainen viiva on parhaiten datapisteisiin sovitettu suora viiva, joka mallintaa dataa parhaiten. [24.] Vaikka sinisten pisteiden paikka heittelee runsaasti kuvassa 3, niiden antama arvo nousee lineaarisen suoran mukaisesti. Punainen suora sovitetaan siten, että se minimoi etäisyyden suoran ja datapisteiden välillä. Suoran antamaa tietoa voidaan käyttää hyväksi ennustettaessa uuden syötettävän datan antamaa tulosta. [17, kpl 1.2.]



Kuva 3. Esimerkki lineaariregressiosta [24].

Lineaariregression mallia luotaessa tehtävässä törmää matemaattisiin funktioihin, joista ensimmäinen on lineaarisen suoran malliin käytettävä funktio, joka näkyy kaavassa 2.

$$y = m * X + b \quad (2)$$

y on funktion ratkaisu ja se kertoo suoran matemaattisen yhtälön

m on suoran kulmakerroin

X on pisteen x -akselin koordinaatit

b on suoran sijainti y -akselilla

Koneoppimisen kannalta lineaariregressiossa luotava lineaarinen suora saadaan kaavalla 2. Tähän käytetään syötettävää dataa ja algoritmissa käytettävää iterointia, jonka avulla etsitään suoran kulmakerroin ja suoran sijainti y -akselilla, jotka ilmentävät dataa parhaiten. Iteroinnilla tarkoitetaan jonkun asian toistoa. Projektissa lineaariregression luonnissa toistetaan kustannusfunktion ja gradient descent -algoritmin käyttöä tarvittavien muuttujien etsinnässä.

Kustannusfunktio arvioi lineaariregression toimintaa. Sen käyttö laskee virhe-eron kaikkien yksittäisten näytekappaleiden osalta ja laskee ne yhteen. Virheellä tarkoitetaan ennustetun dataan ja oikean datan välistä eroa. Kustannusfunktio tekee muodostettavasta mallista tarkan, mikäli sen antama tulos on pieni. Kustannusfunktion yhteydessä puhutaan häviöfunktioista, jolla tarkoitetaan yhden kustannusfunktion laskussa olevan näytteen laskusta. Kustannusfunktion kaava on nähtävissä kaavassa 3.

$$Cost = \frac{1}{2N} \sum_{i=1}^N (Y' - Y)^2 \quad (3)$$

N on näytekappaleiden määrä

Y' on ennustettu lineaarisen suoran malli

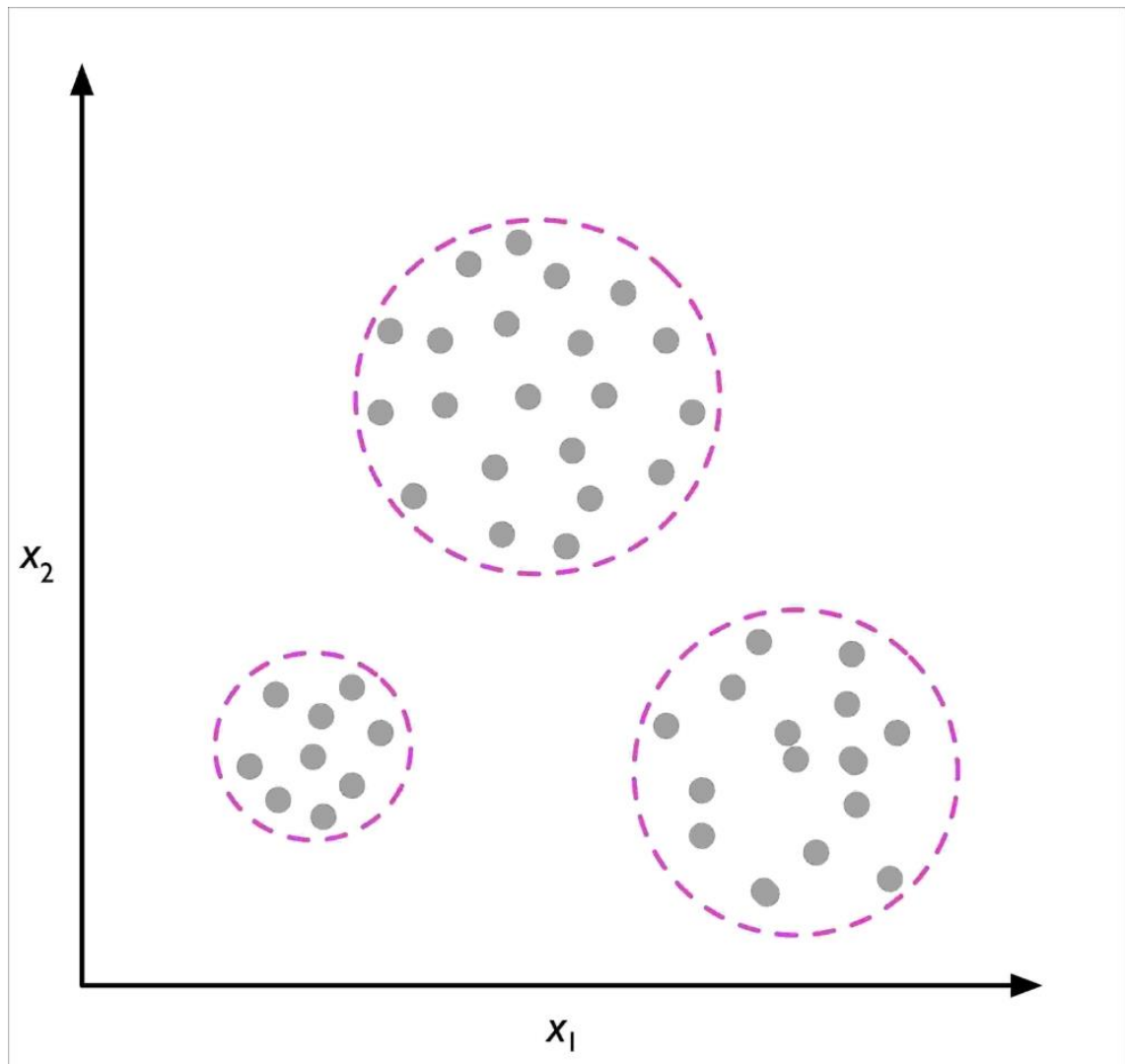
Y on akselille määrätyn datajoukon piste

Gradient descent -algoritmi on funktion pienimmän arvon löytämiseen tarkoitettu optimointialgoritmi. Algoritmin käytön ideana on laskea funktion derivaatta eli kaltevuus, mikä mahdollistaa funktion pienimmän arvon löytämisen. [25.]

4.2.2 Ohjaamaton oppiminen

Ohjaamattoman oppimisen tyyppi eroaa ohjatusta oppimisesta, sillä ohjaamaton oppiminen sisältää nimeämätöntä dataa ja tuntemattomia datarakenteita. Ohjaamattoman oppimisen tehtäväkategoriat tarjoavat tavan käydä läpi datan sisältämää rakennetta tarvittavan datan keräystä varten. Ohjaamaton oppiminen sisältää vain tulodataa mutta ei vastaava lähtödataa, jolloin käyttäjän on pääteltävä mallin rakenteesta tarvittava vastaus [20].

Klusterointi on ohjaamattoman oppimisen tehtäväkategoria, jonka avulla voidaan järjestää datajoukko merkityksellisiin alajoukkoihin, klustereihin, joilla ei ole tietoa joukon varsinaisesta luokittelutavasta. Jokainen systeemiin syntyvä klusteri sisältää joukon dataa, jotka ovat lähellä toisiaan ja sopivat parhaiten siihen klusteriin, joka sijaitsee lähimpänä. [17, kpl 1.2.] Kuvassa 4 on esimerkki kolmesta syntyneestä klusterista, joiden sisältämät datapisteet ovat lähellä toisiaan.



Kuva 4. Esimerkki klusteroinnista [17, kpl 1.2].

4.2.3 Vahvistusoppiminen

Vahvistusoppiminen koostuu tavoitteesta kehittää järjestelmä, joka päivittää itseään ympäristön vuorovaikutuksen avulla. Vahvistusoppiminen sisältää palkintosiinaaleja, jotka

ohjaavat järjestelmää käyttäytymään tietyllä tavalla. Järjestelmä voi oppia sarjan toimintoja vahvistusoppimista käyttäen, sillä jokaisesta oikeasta toiminnosta järjestelmä saa palkintosignaalin. Jokainen oikea toiminto opettaa järjestelmää toimimaan uuden datan kanssa halutulla tavalla. Opettamiseen käytetään joko yritys-erehdys-tapaa tai tarkan suunnittelun mukaista lähestymistapaa. Vahvistumisoppiminen sisältää useampia tehtäväkategorioita, mutta perusajatus kategorioissa on se, että järjestelmä pyrkii maksimoimaan palkintojen määrän ympäristössä tapahtuvien asioiden kautta. Jokainen ympäristön muutos voidaan yhdistää positiiviseen tai negatiiviseen palkintoon, joka vaikuttaa järjestelmän toimintaan. Esimerkkinä voidaan käyttää shakkiottelua, jossa jokainen siirto voidaan lukea ympäristön muutokseksi. Vastustajan nappulan syönti voidaan osoittaa antavan palkinnon, kun taas oman nappulan syönti antaa negatiivisen palkinnon. [17, kpl 1.2.]

4.3 Mallit

Mallit muodostavat pääosa-alueen koneoppimisessa, sillä ne muodostuvat käytössä olevan datan oppimisesta sekä lisäksi ovat vastaus käsillä olevaan tehtävään. Koneoppimismalleja on laaja joukko, johtuen eri koneoppimistehtävistä. Malliryhmiä on kolme, jotka ovat geometriset mallit, todennäköisyysmallit sekä loogiset mallit. [15, s.20, 21.]

4.3.1 Geometrinen malli

Geometrinen malli on rakennettu suoraan ilmentymäavaruuteen käyttäen geometrisiä käsitteitä, kuten suoria, tasoja ja etäisyyksiä. Ilmentymä on synonyymi sanoille esimerkki, esine, tapaus ja kohde. Ilmentymäavaruus on tila kaikille mahdollisille ilmentymille ja sitä kuvaillaan geometriseksi tilaksi, jonka jokainen ominaisuus on yksi tilan ulottuvuuksista. [26, s. 549.] Kuvassa 3 oleva lineaariregressio on yksi esimerkki geometrisestä mallista. Geometriset mallit tarjoavat käyttäjälleen ymmärrettävän näkemyksen mallin toiminnasta ja mahdollistavat mallien muutoksia datan vaihteluilla. [15, s. 21, 25.]

4.3.2 Todennäköisyysmalli

Todennäköisyysmallit perustuvat todennäköisyyksiin eri tekijöiden välillä. Todennäköisyydet perustuvat ennustettavissa oleviin tapahtumiin ja niiden arvioitaviin todennäköisyyksiin. Käytännössä todennäköisyyksiä ennustetaan kaksien muuttujien välille, niiden jonka data tunnetaan ja niiden jonka data on kiinnostuksen kohteena. Yleensä näiden datojen välillä on joku yhteys, joka perustuu tuntemattomaan todennäköisyysjakaumaan. Tähän perustuu mallin muodostus. Kuvassa 5 on esimerkki mallin käytöstä, jossa spam-postiin perustuvassa tapauksessa todennäköisyysjakaumasta syötetyt arvot ennustavat postin olevan spam-postia, mikäli jompikumpi sanoista "Viagra" tai "lottery" esiintyy postissa. Sähköposti ei ole spam-postia, mikäli kumpikaan sanoista tai kumpikin sana esiintyy postissa. "Viagra" ja "lottery" edustavat tunnettua dataa ja luokat "spam" ja "ham" ovat kiinnostuksen kohteena olevaa dataa. Tummennetut luvut edustavat vahvemman todennäköisyyden luokkaa. [15, s. 8, 25, 26.]

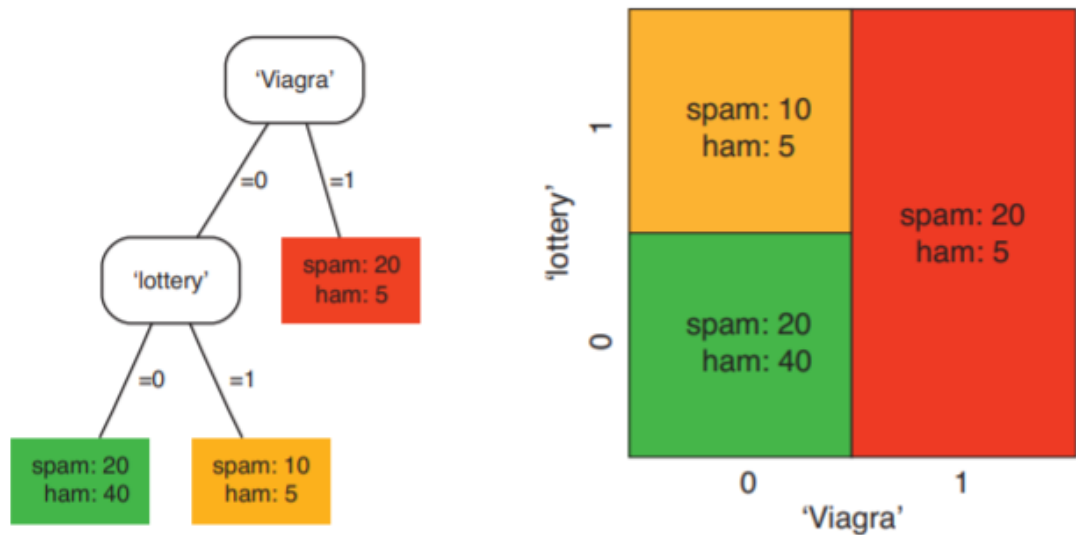
Viagra	lottery	$P(Y = \text{spam} \text{Viagra}, \text{lottery})$	$P(Y = \text{ham} \text{Viagra}, \text{lottery})$
0	0	0.31	0.69
0	1	0.65	0.35
1	0	0.80	0.20
1	1	0.40	0.60

Kuva 5. Spam-postin todennäköisyysjakauma [15, s.26].

4.3.3 Looginen malli

Loogiset mallit perustuvat tietokonetieteisiin ja ne sisältävät ymmärrettäviä sääntöjä, kuten ohjelmointikielissä esiintyvää if-then-rakennetta. Kyseiset säännöt voidaan järjestää puukaavioksi, siten että ylimpänä oleva muuttuja johtaa kahteen oksaan eli kahteen vaihtoehtoon. Kummankin vaihtoehdon päässä on muuttuja, joista johtaa taas kaksi vaihtoehtoa eteenpäin ja näin jatketaan, kunnes saavutaan oksien lehtiin, jossa tapahtuu luokan luokittelu, todennäköisyyden luonti tai esimerkiksi reaaliarvon määrittäminen. Edellä mainittuja puukaavioita kutsutaan yleisesti päätöspuiksi. Loogisen mallin sisältämä data voidaan puukaavioiden lisäksi sijoittaa ilmentymäavaruuteen osittamalla. Kuvassa 6 on esi-

merkki sekä päätöspuusta että ilmentyväävaruuden osittelusta. [15, s. 32,33.] Esimerkissä käytetään samoja muuttujia ja luokkia kuin kuvassa 5 olevassa esimerkissä. Laatikoissa olevat luvut luokkiin "spam" ja "ham" viitaten tarkoittavat sitä, että suuremman luvun luokka vallitsee siinä laatikossa. Luvut kertovat datajoukossa tapahtuneen luokittelun määrästä ja niiden jakaumasta [15, s.32].



Kuva 6. Esimerkki puurakenteesta ja ilmentymäävaruuden osittelusta [15, s.32].

5 Projektissa käytetyt välineet ja ohjelmat sekä työn kulku

Opinnäytetyöprojektissa käytettiin kahta lämpötila-anturia, yhtä hiilidioksidianturia ja yhtä pinnanmittausanturia mittaamaan dataa koneoppimisalgoritmia varten. Anturit kytkettiin Siemensin SIMATIC-automaatiojärjestelmän ET 200SP -sarjan PLC-logiikkaan (Programmable Logic Controller). Anturien syöttämä tieto kulkeutui MindConnect Nano -laitteen kautta MindSphere-IoT-rajapintasovellukseen. MindSphere-sovelluksessa ollut data tuotiin .csv-tiedostomuodossa ulos sovelluksesta. Tiedostojen sisältämää dataa muokattiin Microsoft Excel -ohjelmalla. Lopuksi koneoppimissovellus tehtiin Anaconda-jakelualustalla Python-ohjelmointikielellä.

5.1 Anturit

5.1.1 Pt100-anturi ja sen tarvitsema lähetin

Projektissa Pt100-anturina toimi Danfossin MBT 153 -anturi, jossa on 4 johdinta pitäen sisällään kahta Pt100-anturia. Projektissa käytettiin yhtä Pt100-anturia. Pt100-anturia käytetään lämpötilan mittaukseen ja se luokitellaan RTD:ksi (Resistance Temperature Detector). Pt100 koostuu platinasta, johon viittaa sanan alkuosa Pt. Platinan käytön edut ovat lämpötilan noustessa tai laskiessa anturin resistanssin lineaarisuus sekä arvon pysyminen vakaana korkeissakin lämpötiloissa. Luvulla 100 viitataan anturin resistanssin nimelliseen arvoon lämpötilan ollessa 0 °C. [27.] Pt100-anturin resistanssin arvo nousee ja laskee lineaarisesti lämpötilan arvon noustessa ja laskiessa. Kuvassa 7 on kuva anturista.



Kuva 7. PT100-anturi.

Pt100-anturin antama ulostulo on resistanssiarvo, jota esimerkiksi projektissa käytetty PLC-logiikka, Siemensin valmistama ET 200SP, ei pystynyt suoraan lukemaan, koska

PLC-logiikat yleisesti lukevat vain 4–20 mA:n virtaviestejä tai 0–24 VDC:n jänniteviestejä. Tämän takia PT100-anturin kanssa käytettiin Jumon dTrans 03 -lähetintä, joka muuntaa Pt100-anturin resistanssiarvon virtaviestiksi. Lähettimeen voidaan kytkeä kaksi- tai kolmejohtiminen Pt100-anturi. Lähettimen ulostyöntämä jännite- tai virtaviesti on lineaarinen resistanssin kanssa. Lähettimellä on lyhyt reaktioaika. [28.] Jännite- tai virtaviesti muunnetaan PLC-logiikkaan syötetyn ohjelman välityksellä lämpötilaksi. Kuvassa 8 on lähettimen kuva.



Kuva 8. dTrans 03 -lähetin.

5.1.2 EE800 CO₂- ja lämpötilälähetin

EE800-lähetin on kiinteistöautomaation sovelluksiin käytettävä laite, joka mittaa CO₂-pitoisuutta ja lämpötilaa tilasta, jossa laite sijaitsee. Laitteen CO₂-anturi tasaa ajan myötä mittausarvoaan ja anturin mittausarvo ei heittele saasteiden vuoksi. Lähettimessä on

myös passiivinen lämpötila-anturi, jonka tyyppiä ei mainita lähettimen oppaassa. Lähetin on varustettu upotetulla digitaalisella näytöllä [29.] EE800-lähetin on kuvattuna kuvassa 9.



Kuva 9. EE800-lähetin.

Lähettimen CO₂-anturi on NDIR-anturi (Non Dispersive Infra-Red). NDIR tarkoittaa käsitettä, jossa infrapunasäde ei hajoa valon lähettäjän ja vastaanottajan väliseen väliaineeseen, kun valoa syötetään valon lähettjästä. NDIR-menetelmällä toimivalla anturilla voidaan mitata esimerkiksi kaasun pitoisuutta säiliöstä. Infrapunavalon absorboituminen kaasuun ja anturin vastaanottajan saavuttava infrapunavalo riippuvat kaasun määrästä valonlähettäjän ja vastaanottajan välillä. CO₂-anturissa hiilidioksidi pääsee liikkumaan vapaasti valon lähettäjän ja vastaanottajan välillä. Anturin vastaanottajassa on suodatin, joka tunnistaa tulevan valon elektromagneettisen spektrin aallonpituuden perusteella. CO₂-pitoisuutta mitataan yksiköllä ppm (parts per million). [30.]

5.1.3 Kapasitiivinen lähestymisanturi E2K-F10MC1

Projektin pinnankorkeusanturina toimi kapasitiivinen lähestymisanturi E2K-F10MC1. Sen ominaisuuksiin kuuluu aineiden ja esineiden tunnistaminen. Toimintaetäisyys anturilla on maksimissaan 10 mm, mutta etäisyys vaihtelee riippuen kohteen sähkönjohtavuudesta ja vesipitoisuudesta. Anturin antama ulostulojännite on 1,5 VDC. [31.] Anturi on nähtävissä kuvassa 10.



Kuva 10. Kapasitiivinen lähestymisanturi.

Anturi on NO-tyyppinen eli aktivoituessaan antaa ulos 1,5 VDC. Tämä jännite ei riittänyt projektissa PLC-logiikan digitaalisen tulon aktivoitumiseen, vaan jännitteen piti olla vähintään 11 VDC [32, s.34]. Ratkaisuna ongelmaan käytettiin MJE3055T-transistori. MJE3055T-transistorin tarkoituksena on olla virtapiirien kytkimenä ja vahvistimena. Transistori ottaa vastaan Base-napaan korkeintaan 5 V ja pystyy välittämään Collector-navasta Emitter-napaan korkeintaan 60 V, kun Base-napa on jännitteellinen. [33.] Lisälaitteena transistorin ja anturin kanssa toimi välirele, joka pystyi välittämään 24 VDC:n jännitettä apukoskettimiltaan eteenpäin vetävässä tilassa.

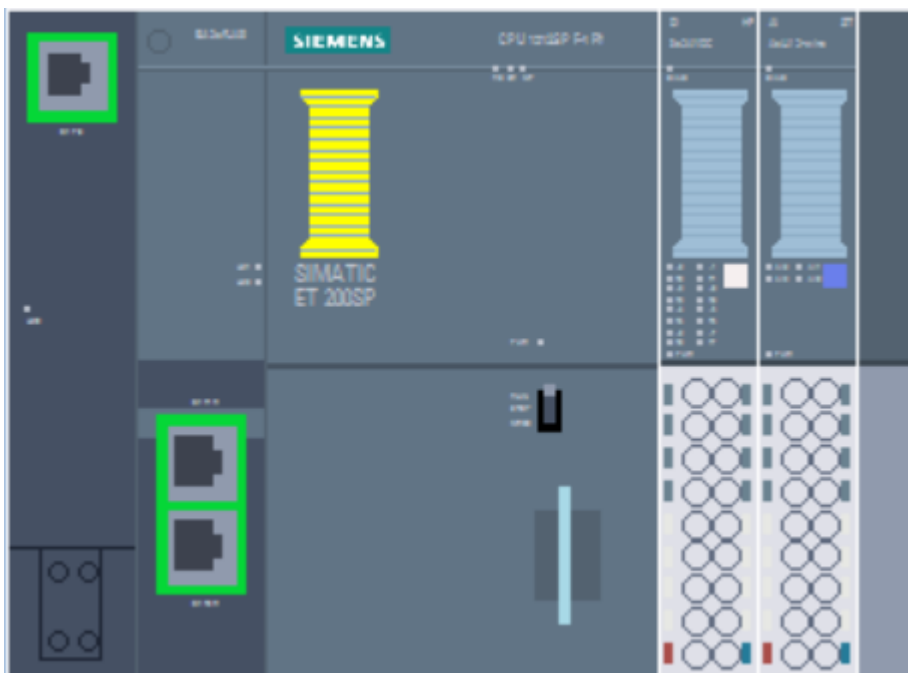
5.2 PLC-logiikka ja sen käyttämät moduulit

Projektissa käytettiin SIMATIC ET 200SP -logiikkaa ja siihen liitettäviä lisälaitteita, kuten digitaalisia ja analogisia tulokortteja. Nämä kortit kiinnitettiin logiikan tiedon välittäjinä toimiviin BaseUnit-moduuleihin. Logiikan toisena päänä toimi server-moduuli.

5.2.1 SIMATIC ET 200SP - CPU 1512SP-1 PN

Projektissa käytettiin SIMATIC:n ET 200SP -PLC-logiikkaa, jonka keskustietokoneena (CPU) toimii malli CPU 1512SP-1 PN. CPU-keskustietokone sopii vaatimuksiltaan keskikokoisiin sovelluksiin käsittelykykynsä ja vastauskykynsä perusteella. CPU-keskustietokonetta voidaan käyttää PROFINET IO -verkossa joko pää-PLC- tai etä-PLC-logiikkana. [34.] PROFINET IO on Ethernet-pohjainen teollisuuden käytössä oleva verkkoprotokolla, jota esimerkiksi Siemens käyttää eri laitteiden tiedonsiirrossa [35]. PROFINET IO -verkolla voidaan hajauttaa tiedonkeruu automaatiojärjestelmästä moneen eri PLC-logiikkaan ja CPU 1512SP-1 PN voi toimia joko kaiken keräävänä pää-PLC-logiikkana tai yhtenä PROFINET IO -verkon etä-PLC-logiikoista, jolloin se lähettää keräämänsä datan pää-PLC-logiikalle.

CPU 1512SP-1 PN toimii 24 VDC:n jännitteellä, siinä on SIMATIC-muistikorttipaikka ja yksi RJ45-liitinpaikka PROFINET IO -verkkoa varten. Lisäksi CPU-keskustietokoneeseen on mahdollista liittää BusAdapter-lisämoduuli, jolla on mahdollista saada lisää yhteysmuotoja PLC-logiikasta esimerkiksi hajautettua järjestelmää varten. SIMATIC-muistikortti on pakollinen, sillä ilman sitä CPU-keskustietokone ei toteuta käskyjä. [34.] Kuvassa 11 on nähtävissä Siemensin TIA Portal -ohjelman laitenäkymä projektissa käytetystä logiikasta. TIA Portal on Siemensin PLC-logiikoiden ohjelmointiympäristö. Kuvassa on nähtävissä keskustietokone suurena palikkana, joka vie kaksi kolmasosaa kuvasta. Kuvan oikealla puolella on kaksi lisämoduulia kiinnitettynä BaseUnit-moduuleihin ja kuvan äärioikeassa laidassa oleva palikka on server-moduuli.



Kuva 11. TIA Portal -projektin laitenäkymä.

5.2.2 Server-moduuli ja BaseUnit-moduulit

ET 200SP -PLC-logiikka koostuu edellisessä kappaleessa käsitellystä CPU-keskustietokoneesta sekä halutusta määrästä lisämoduuleja. BaseUnit-moduulit toimivat pohjana logiikkaan lisättäville lisämoduuleille. BaseUnit-moduuli kiinnitetään CPU-keskustietokoneeseen, jonka sivussa on aukot BaseUnit-moduulin kytkennälle. BaseUnit-moduulissa on väkäset, jotka sopivat CPU-keskustietokoneen sisälle. Lisäksi jokaisen BaseUnit-moduulin vastakkaisella sivulla on vastaavanlaiset aukot kuin CPU-keskustietokoneessa, joten BaseUnit-moduuleja voidaan laittaa sarjaan haluttu määrä. Moduuleja voidaan laittaa peräkkäin jopa 64 kappaletta [34].

BaseUnit-moduulit välittävät ET 200SP -PLC-logiikan lisämoduulien sähköiset ja mekaaniset yhteydet. BaseUnit-moduulissa on lisämoduulipaikan lisäksi vaihteleva määrä kiinnitettävien johtimien terminaalipaikkoja. Terminaalipaikkojen määrä vaihtelee BaseUnit-typin mukaan, joita on useita erilaisia. A0 BaseUnit-moduulissa, jota käytettiin projektissa, on 16 terminaalipaikkaa esimerkiksi anturikytkentöjä sekä BaseUnit-moduulin 24 VDC:n jännitteen plus- ja maa-napoja varten. BaseUnit-malleja on erivärisiä, jotka vaikuttavat 24 VDC:n jännitteen kytkentöihin. Valkoinen malli käynnistää uuden ryhmän,

jonka jälkimmäiset harmaat moduulit eivät tarvitse 24 VDC:n jännitekytkentöjä, sillä valkoinen moduuli syöttää jännitteen harmaille moduuleille. [36, s.21, 23.] ET 200SP -logiikassa pitää olla ainakin yksi valkoinen moduuli, jotta jännitteensyöttö onnistuu BaseUnit-moduulien kautta lisämoduuleille. Projektissa käytettiin valkoisen mallin BaseUnit-moduuleita.

Server-moduuli on ET 200SP -logiikan lisämoduulien jälkeen viimeiseksi sijoitettava osa, joka täydentää logiikan kokonaisuuden. Sen tehtävä on sulkea logiikalta BaseUnit-moduulien läpi kulkeva väylä. Lisäksi se sisältää kolme varasulaketta. [37, s.8.]

5.2.3 Analoginen ja digitaalinen lisämoduuli

ET 200SP -logiikan BaseUnit-moduulien tarkoitus on mahdollistaa lisämoduulien asennus PLC-logiikan yhteyteen. Lisämoduulit asennetaan BaseUnit-moduuleissa oleviin sopiviin aukkoihin. Tässä projektissa käytettiin sekä analogista ja digitaalista input-moduulia, joiden lisääminen mahdollistivat projektin anturien lisäyksen PLC-logiikkaan. ET 200SP Analog Input Module AI 4xU/I 2-wire ST -lisämoduuli mahdollistaa neljän analogisen tulon asennuksen. TIA Portal -ohjelmointiympäristössä moduulin voi määrittää joko mittaamaan tulojen jännitettä (U) tai kaksijohtimisen lähtetimen antamaa virtaa (mA). [38, s.12.] Digital input module DI 8x24VDC HF -moduulin avulla PLC-logiikkaan voidaan asentaa 8 digitaalista signaalia vastaanottavaa tuloa [32, s.12].

5.3 TIA Portal -ohjelmisto

TIA Portal (Totally Integrated Automation) on Siemensin luoma automaation ohjelmointiympäristö, jonka tarkoituksena on yhden ohjelman sisällä tapahtuva PLC-ohjelmointi, verkkotopologian määrittely ja käyttöliittymien suunnittelu [39, s.12]. TIA Portal -ohjelmiston sisälle Siemens yhdisti aiemmin heidän luomansa SIMATIC STEP 7- ja SIMATIC WinCC -suunnitteluohjelmistot [40]. TIA-lyhenteellä tarkoitetaan automaation yhdistettyä konfigurointia, ohjelmointia, datan varastointia ja datan siirtoa.

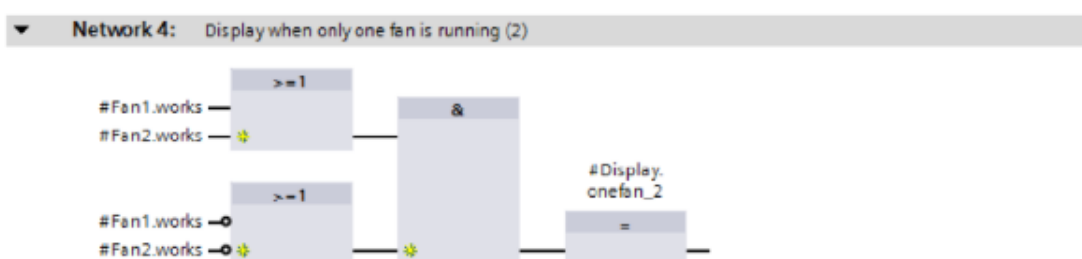
5.3.1 SIMATIC-käsite ja TIA Portal -ohjelmiston sisältö

SIMATIC-käsitteellä tarkoitetaan Siemensin luomaa SIMATIC-automaatiojärjestelmää, jonka laitteet pystyvät kommunikoimaan keskenään erilaisten yhteydenpitoväylien avulla. Laitteisiin kuuluvat muun muassa SIMATIC PLC -logiikkaohjaimet, SIMATIC HMI -asemat, SIMATIC NET, joka toimii laitteiden välisenä yhteydenpitoverkkona sekä SIMATIC STEP 7 -ohjelmisto. SIMATIC STEP 7 on suunnitteluohjelmisto, jonka avulla voidaan määrittää ja ohjelmoida SIMATIC-järjestelmän komponentteja. TIA Portal -ohjelmiston modernisoidun STEP 7:n tärkeimmät ominaisuudet ovat käyttöön otettavan järjestelmän virtuaalinen määrittely, yhteysmuotojen määrittely ja asetukset eri laitteiden välillä sekä käyttäjäohjelman ohjelmointi haluttuun PLC-logiikkaohjaimeseen. Käyttäjäohjelmaa luotaessa logiikan ohjelmointikielien vaihtoehtoina ovat tikapuukaavio (LAD), function block diagram (FBD), statement list (STL) ja structured control language (SCL). [41, s. 9–11.] SIMATIC WinCC (Windows Control Center) on HMI-käyttöliittymälaitteiden ja -sovellusten suunnittelu- ja käynnissäpito-ohjelmisto [41, s.293]. TIA Portal -ohjelmistosta löytyy useampia eri versionumeroinnin perusteella nimettyjä ohjelmaversioita. Projektissa käytettiin TIA Portal V16 -ohjelmistoa.

5.3.2 Function Block Diagram -ohjelmointikieli

Projektin aikana tehdyn ja PLC-logiikkaan syötetyn ohjelman ohjelmointikielenä oli FBD eli function block diagram. FBD on graafinen ohjelmointikieli ja se soveltuu ohjelmoitavien logiikoiden ja ohjainten ohjelmoimiseen. FBD-ohjelmointikieli sisältää function block- ja function-elementtejä, jotka toteuttavat käskyjä ja esimerkiksi if-then-lauseerakennetta. Elementit voidaan yhdistää signaalinkulkuviivoilla elementin lähtöpuolelta, joka on elementin oikeapuoli, seuraavan elementin tulopuolelle, joka on elementin vasen puoli. [42, s. 212, 221.] Siemensin STEP 7 -ohjelma sisältää laajat ohjelmakirjastot, joiden avulla luotavan ohjelman muuttujat voidaan lisätä esimerkiksi binääristen AND- ja OR-logiikkaelementtien, ajastimien, laskureiden tai muun muassa matematiikkaelementtien yhteyteen. STEP 7 -ohjelmalla voidaan luoda myös omia function-elementtejä, joilla kopioidaan ja yksinkertaistetaan ohjelman luontia.

Function block diagram -ohjelmointikielellä toteutettu ohjelma koostuu yhdestä tai useammasta ohjelmalohkosta. Ohjelmalohko taas koostuu yhdestä tai useammasta network-osasta. Network-osa sisältää ainakin yhden loogisen toiminnon, joka voi olla yksinkertainen binäärinen päälle/pois-toiminto tai se voi sisältää monimutkaisemman toiminnon. [43, s. 323.] Ohjelmalohkojen pitää olla linkittyneitä toisiinsa. Tavallisesti tämä tehdään yhdellä Main-ohjelmalla, johon yhdistetään käytössä olevat ohjelmalohkot. Kuvassa 12 on esimerkki yhdestä network-osasta. Siinä on kaksi OR-function block -elementtiä ja lisäksi yksi AND- ja yksi Assignment-function block -elementti. Lisäksi alemman OR-elementin tulopuolella on kaksi palloa, jotka tekevät negaation tulopuolelle. Oikealla oleva lähtö käynnistyy, kun jompikumpi muuttujista "Fan1.works" tai "Fan2.works" on aktivoitunut.



Kuva 12. Esimerkki FBD-ohjelmasta [42, s.329].

5.4 MindSphere

MindSphere on Siemensin avoin IoT-rajapintasovellus (Internet of Things), joka toimii pilvipohjaisesti selaimessa. MindSpheren avulla käyttäjät pystyvät kytkemään käytössään olevia fyysisiä, Internet- ja yrityspohjaisia järjestelmiä yhden sovelluksen alle. MindSphere-sovellus toimii käyttäjärjestelmänä ja se sisältää Siemensin omien asiantuntijoiden tarjoamia erilaisia sovelluksia, joihin lukeutuvat muun muassa MindSpheren omat ohjelmat Fleet Manager ja Asset Manager. Nämä ohjelmat tarjoavat käyttäjälleen datan tuonti- sekä analytiikka- ja visualisointipalvelun, mitkä mahdollistavat käyttäjälleen tunnuslukujensa tehokkaan tarkkailun. MindSphere sisältää myös muita ohjelmia teollisten elementtien tarkasteluun ja niiden analysointiin. Lisäksi MindSphere tarjoaa myös kehittäjäympäristön, johon kehittäjät voivat tuoda omia sovelluksiaan. [44, s.4, 5, 9.]

MindSphere on PaaS-muotoinen (Platform as a Service) palvelu. Käsitettä käytetään palvelimien virtualisoinnin ja pilvipalveluiden yhteydessä ja sillä tarkoitetaan käytettävän palvelun tarjoavan käyttäjälleen palvelun palvelimet, tilan eli muistikapasiteetin ja käyttöjärjestelmän yhtenä osana palveluaan. Käyttäjä itse joutuu tuomaan palveluun käsiteltävän datan sekä sovellukset. [45.] MindSpheren yhteydessä PaaS-käsitteellä tarkoitetaan, että markkinoiden johtavat pilvipalveluiden tarjoajat, kuten Amazon ja Microsoft, isännöivät MindSpheren tarjoamia palveluita.

MindSphere-sovelluksen avulla käyttäjä pystyy yhdistämään järjestelmissään olevat muuttujat MindSphereen. Siemensin käyttämä nimitys yhteyden tarjoajasta on MindConnect. [44, s.5.]

MindConnect Nano on Siemensin valmistama laite, jonka avulla voidaan luoda yhteys MindSpheren ja käyttäjän järjestelmien välille. Laite myös mahdollistaa käyttäjän datan siirron MindSphereen. Datan siirtoon ja keräykseen MindSpheren ja käyttäjien järjestelmien välillä käytetään Siemensin omaa S7- ja OPC UA -protokollaa. Laite tukee yhteyden luontiin käytettävää turvattua Internet-yhteyttä. [46]. Kuvassa 13 on MindConnect Nano -laite.



Kuva 13. MindConnect Nano -laite [47, s.18].

5.5 Koneoppimiseen käytettävät menetelmät

Tässä luvussa käsitellään koneoppimissovelluksen luontiin käytettyä ohjelmointikieltä, ohjelmaa sekä sovelluskirjastoa.

5.5.1 Python

Python on oliopohjainen ohjelmointikieli, joka julkaistiin käyttöön vuonna 1991. Sen kehittäjä on alankomaalainen Guido van Rossum. Oliopohjaisuus tarkoittaa sitä, että kielen käyttö perustuu oliopohjaiseen suunnitteluun ja sen käyttöön toimivassa järjestelmässä. [48, kpl 1.] Oliopohjaisuus on luokan ilmentymä. Esimerkiksi luotu luokka "Auto" pitää sisällään erilaisia ominaisuuksia, kuten väri, malli, vuosi ja ovien määrä. Luokan sisälle luodaan ominaisuuksien lisäksi käyttäytymistapoja, jotka periytyvät ominaisuuksien mukana ilmentymille. Oliopohjaisuus ottaa siis käyttöön oliota luodessa luokan ominaisuudet ja käyttäytymistavat. Ilmentymässä ominaisuudet eli luokan muuttujat saavat sisällön, jolloin muuttujat väri ja vuosi saavat sisällön, kuten esimerkiksi "sininen" ja 1980. [49.]

Python on yksi maailman suosituimmista ohjelmointikielistä ja sitä edistävät sen käyttö opetuksessa, sen avoin lähdekoodi ja käytön nopea oppiminen verrattuna muihin ohjelmointikieliin. Oliopohjaisuutta alleviivaavat Python-kielissä olevien kirjastojen käyttö, jotka mahdollistavat merkittävien tehtävien luonnin ilman suurta koodin luontityötä. [48, kpl 1.]

5.5.2 TensorFlow

Yksi Python-luvussa mainituista kirjastoista on koneoppimiseen erikoistunut alusta TensorFlow. Se sisältää työkalut, joiden avulla koneoppimisen tutkijat ja kehittäjät pystyvät luomaan koneoppimissovelluksia. TensorFlow-alustan luoja oli Google Brain -työryhmä Googlen Machine Intelligence Research -järjestöstä, joiden tarkoitus oli suorittaa koneoppimisen ja syvien hermoverkkojen tutkimista. Näin he tulivat luoneeksi samalla TensorFlow-alustan. [50.]

TensorFlow-alusta on koneoppimisalgoritmien esittämiseen ja toteuttamiseen soveltuva rajapinta. Alustan avulla esitettävä laskutoiminta voidaan suorittaa monilla järjestelmillä

kannettavista laitteista mittaviin keskitettyihin laskutoimintaa edistäviin tietokonerykelmiin. Alustan avulla voidaan esittää esimerkiksi opetus- ja päättelyalgoritmeja syvien hermoverkkojen malleille. TensorFlow-alustaa on hyödynnetty esimerkiksi puheentunnistuksen, konenäön ja robotiikan sovelluksissa. [51.]

5.5.3 Anaconda

Anaconda on avoimen lähdekoodin jakelualusta, jonka avulla voidaan työskennellä datatieteiden ja koneoppimisen sovelluksien kanssa. Anaconda-jakelualusta tukee Python-ohjelmointikieltä. Anaconda-jakelualusta sisältää pakkauksienhallinnan ja -asennuksen, ympäristönhallinnan, datatieteiden monikäytön sekä yli 7500 avoimen lähdekoodin pakettia. [52; 53; 54.]

Anaconda-jakelualusta helpottaa sovelluksien tekoa, sillä Anaconda-jakelualusta tuo valmiiksi suuren määrän kirjastoja tarjolle. Lisäkirjastojen asennus onnistuu Anaconda-jakelualustan sisältämän Python-tulkin ulkopuolella. Python-tulkissa tarvitsee vain tuoda kirjastot käyttäjän luomaan koodiin. Anaconda-jakelualusta tarjoaa komentokehoteohjelman, jossa käyttäjä pystyy luomaan useamman ympäristön eri tarkoituksia varten, käyttämään tekstipohjaista Python-tulkkia ja asentamaan eri paketteja. Komentokehoteohjelman lisäksi käyttöön on mahdollista ottaa graafinen käyttäjärajapinta Navigator, joka mahdollistaa eri ohjelmien asennuksien ilman komentojen kirjoitusta [55]. Ohjelmat mahdollistavat muun muassa graafisen tarkkailutavan eri sovellusten kehitykseen.

Yksi Navigator-käyttäjärajan tarjoamista sovelluksista on Jupyter Notebook, joka on Project Jupyterin ylläpitämä avoimen lähdekoodin verkkosovellus. Sovelluksella pystyy luomaan ja jakamaan luotua koodia, tekstiä ja visualisoituja kuvaajia. Jupyter Notebook toimii Python-kielen kanssa. Navigator-näkymästä avattavan Notebook-sovelluksen avaus tarjoaa selainnäkyvän, joka on samalla Notebook-sovelluksen palvelin. [56.] Palvelin-näkymästä pystyy avaamaan uuden Notebook-muistion, joka tarjoaa tekstityökalun, ohjelmointikielen työstömahdollisuuden sekä luotavan koodin ajon ja testauksen.

5.6 Työosuuden kulku

Projektin aloituksessa sen raameiksi esitettiin seuraavat kohdat:

- aeroponisen perunanviljelysäiliöille sopivien antureiden kartoittaminen ja niiden hankkiminen
- projektissa käytettävän PLC-logiikan ja sen lisämoduulien kartoittaminen ja hankkiminen
- datan keräystä IoT-moduulin välityksellä pilvipalveluun ja pilvessä olevan datan käyttäminen käyttöön tarkoitettuun koneoppimissovellukseen.

Perunanviljelysäiliön anturoinnin tarkastelu käynnistettiin Urban Farm Labin tiloissa palaverilla, jossa projektin sisältöä käytiin läpi Urban Farm Labin tiloissa toimivien työntekijöiden kanssa. Perunanviljelysäiliön sen hetkinen anturointi ja laajennustarve kirjattiin ylös, jonka pohjalta aloitettiin projektissa tarvittavien antureiden kartoittaminen.

5.6.1 Anturien ja PLC-logiikan kartoitus

Kartoittamisen lähtökohtana oli tutkia edullisia vaihtoehtoja nykyisten antureiden tilalle sekä kartoittaa antureita nykyisten lisäksi parantamaan prosessin käytettävyyttä. Antureiden tarkastelu ja kartoitus tehtiin ensisijaisesti tutkimalla prosessiautomaatiolaitteiden valmistajien Internet-sivuja. Vaihtoehtoja tutkittaessa päädyttiin lopulta tilaamaan toimittaja Elfa Distrelec Oy:ltä projektin käyttöön tulevat anturit. Kartoittaessa tarvittavia antureita otettiin huomioon aeroponisessa perunankasvatusprosessissa tarvittavat arvot, kuten ravinneliuoksen sähkönjohtavuus, ympäristön ja ravintoliuoksen lämpötila ja ympäristön hiilidioksidipitoisuus. Taulukossa 2 näkyy asiakkaalle lähetetty anturilista.

Taulukko 2. Anturit ja hinta.

Nimike	Toimittaja	Listahinta
Sähkönjohtavuusanturi	elfadistelec	172,20 €
Hiilidioksidianturi, kosteus, lämpötila	elfadistelec	205,70 €
Virtausanturi	elfadistelec	76,30 €
Nesteen lämpötila-anturi	elfadistelec	13,80 €
Pinnankorkeusanturi	elfadistelec	68,20 €

Antureiden kartoituksen yhteydessä tutkittiin myös projektissa käytettävän PLC-logiikan hankintaa, johon tilattavat anturit tulisi liittää. PLC-logiikan hankintapäätös perustui projektin asiakkaan toiveeseen, jonka perusteella logiikan päävaihtoehdoiksi valikoituivat valmistaja Siemensin SIMATIC ET 200SP- ja SIMATIC S7-1200 -logiikat. Vaihtoehtojen välillä tutkittiin niiden edullisuutta ja käytännöllisyyttä projektissa sekä PLC-logiikan yhteyteen tarvittavia lisämoduuleja. Vaihtoehdoksi valittiin SIMATIC ET 200SP -PLC-logiikka ja siihen tarvittavat lisämoduulit, jotka tilattiin asiakkaan toimesta Siemensiltä.

PLC-logiikan yhteydessä asiakas tilasi Siemensiltä MindConnect Nano -laitteen ja käyttötilin Siemensin ylläpitämään MindSphere-IoT-rajapintasovellukseen. Sovelluksen käyttöoikeus ja MindConnect Nano -laite mahdollistivat datan siirron PLC-logiikan kautta MindSphere-sovellukseen.

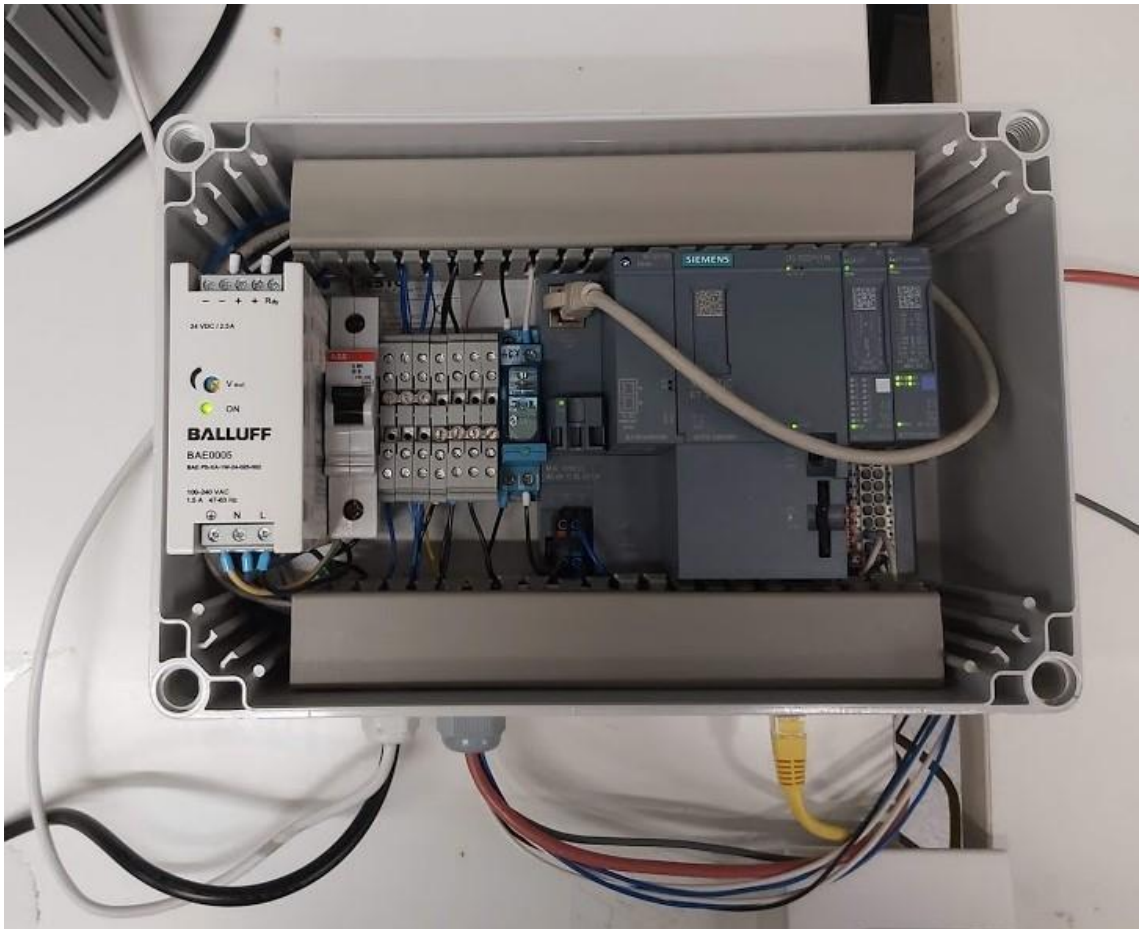
5.6.2 Projektissa suoritettu asennus asennuskoteloon

Tilatuista tavaroista osa asennettiin sähkötukku SLO:lta tilattuun Enston Cubo C -asennuskoteloon. Asennuskotelon sisälle sijoitettuun DIN-kiskoon kiinnitettiin taulukossa 3 näkyvät laitteet. Taulukossa mainitaan laitteiden käyttötarkoitukset, mallit, valmistajat ja määrät. Osa asennetuista tavaroista saatiin projektiin lainakäyttöön.

Taulukko 3. DIN-kiskoon asennetut osat.

Käyttötarkoitus	Malli	Valmistaja	Kappalemäärä
24VDC-jännitelähde	BAE0005	Balluff	1
Johdonsuojakatkaisija jännitelähteen syötölle (230 VAC)	S261-B6	ABB	1
Riviliitin (2-kerrok.)	-	-	7
Relekanta	95.63	Finder	1
Välirele	40.31	Finder	1
PLC-logiikka	SIMATIC ET 200SP	Siemens	1
Digitaalinen input-moduuli	DI 8x24 VDC HF	Siemens	1
Analoginen input-moduuli	AI 4xUI/2-wire ST	Siemens	1
BaseUnit-moduuli lisämoduulien kiinnittämiseen	BaseUnit type A0	Siemens	2
Muistikortti	SIMATIC Memory Card	Siemens	1

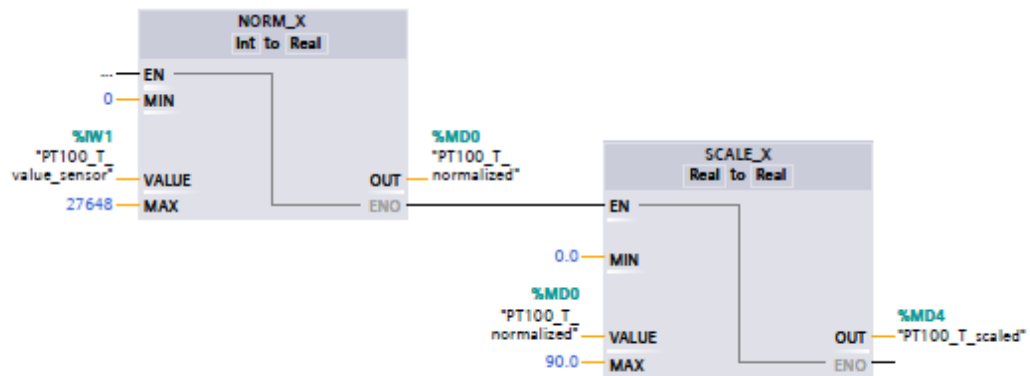
Asennuskotelon sisälle asennettiin taulukossa 3 näkyvien osien lisäksi kaksi kappaletta johtokouruja sekä lähetin muuntamaan PT100-anturin resistanssiarvot virtaviestiksi (4–20 mA). Asennuskoteloon tehtiin läpiviennit anturien syöttö-, maa- sekä signaalijohdoille, jännitelähteen syöttöjohdolle (230VAC), MindConnect-laitteen syöttöjohdolle (24VDC) sekä RJ45-liitinpaikalle PROFINET IO-verkkoa varten. Kuvassa 14 näkyy asennuskotelon sisältö. Asennuskoteloon tehtiin johdotukset riviliitinten, PLC-logiikan tulonastojen ja anturien välille. Lisäksi kapasitiivisen lähestymisanturin piiriin asennettiin transistori MJ3055T.



Kuva 14. Asennuskotelo.

5.6.3 PLC-ohjelman luonti

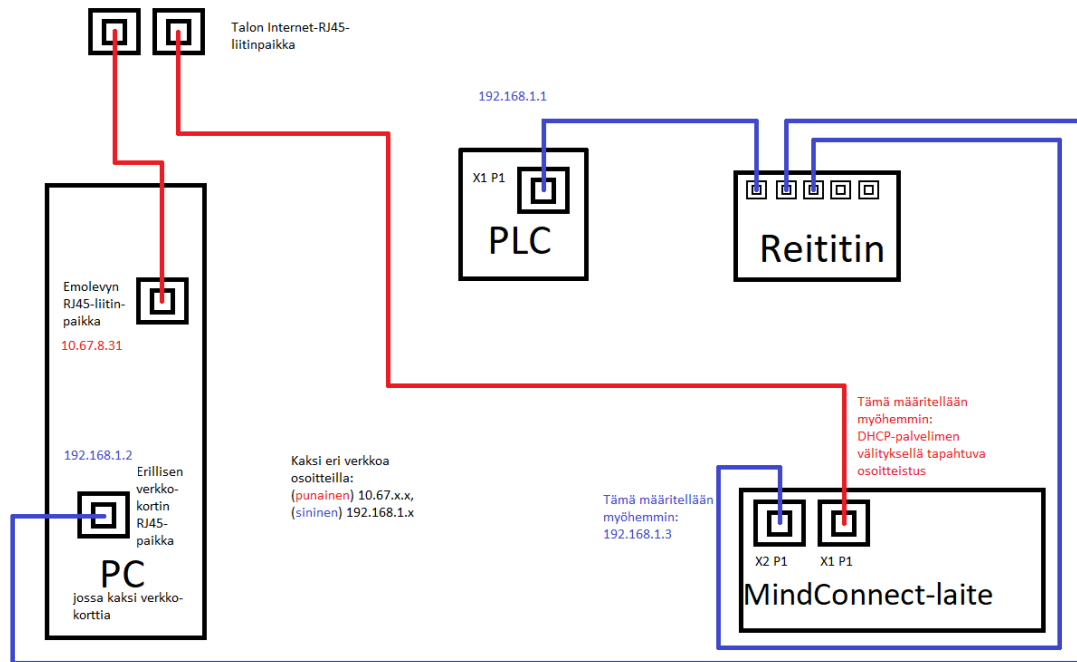
SIMATIC ET 200SP -PLC-logiikalle syötettiin logiikkaohjelma, joka tehtiin Siemensin TIA Portal -suunnitteluohjelmalla FBD-ohjelmointikielellä (Function Block Diagram). Kytkenöissä asennettiin neljä anturia, joiden antaman datan syötön kulku PLC-logiikalle suunniteltiin TIA Portal -ohjelmalla. Kuvassa 15 näkyy FBD-kielellä ohjelman osa käskyistä. Analogiset anturit syöttivät PLC-logiikalle virta- tai jännitearvoa, jotka piti skaalata ohjelman kautta näyttämään suurearvoa. Digitaaliselle tulopuolelle kiinnitettiin nestein pinnan mukaan aktivoituva kapasitiivinen lähestymisanturi, joka ohjelmoitiin näyttämään tilaansa sen aktivoitumisen mukaan. PLC-logiikka otettiin käyttöön syöttämään reaaliaikaista dataa antureilta.



Kuva 15. Osa PLC-logiikalle syötetystä ohjelmasta, jossa PT100-anturin tuottama tieto skaalataan luettavaksi ymmärrettävään muotoon.

5.6.4 Verkkojen luonti

Verkkoyhteydet otettiin laitteiden välillä käyttöön siten, että PLC-logiikka, MindConnect Nano -laitteen toinen RJ45-liitinpaikka ja tietokoneen verkkokortti yhdistettiin samaan lähiverkkoon. Lähiverkkoon ei liitetty yhteyttä Ethernet-verkkoon. Tietokoneen emolevykortti ja MindConnect Nano -laitteen ensimmäinen RJ45-liitinpaikka yhdistettiin Ethernet-verkkoon. Tällä tavoin saatiin erotettua projektin sisäinen verkko omaksi lähiverkoksi, jota MindConnect-laite sekä PLC-logiikka tarvitsivat tiedonsiirtoa varten. Lisäksi MindConnect-laite tarvitsi Ethernet-yhteyden MindSphere-IoT-rajapinnan ja MindConnect-laitteen välistä tiedonsiirtoa varten. Kuvassa 16 näkyy suoritettujen lähiverkon sekä PROFINET IO -verkon asennus. Reitittimenä PROFINET IO -verkossa toimi kytkin ZyxEL GS-105B v3.



Kuva 16. Suoritettavat verkkoasennukset.

5.6.5 Yhteyden luonti MindSphere-sovellukseen ja datojen siirto

Verkkojen luonnin jälkeen aloitettiin MindSphere-sovelluksen käyttöönotto. Aluksi MindSphere-sovelluksessa luotiin asset MindConnect Nano -laitteelle. Assetin luonti mahdollistaa yhteydenpidon MindSphere-sovelluksen ja MindConnect Nano -laitteen välillä ja se sisältää logiikan IP-osoitteen lisäyksen, verkkoprotokollan ja PLC-logiikan muuttujien määrittämisen.

Assetin luonnin yhteydessä luotiin aspect- ja type-profiilit MindSphere-sovelluksen sisälle, mitkä identifioivat luotavan yhteyden kohteena olevaa prosessia MindSphere-sovelluksen sisällä. Lisäksi MindSphere-sovellus tarvitsee tiedon kohteena olevasta datan lähettäjistä, tässä tapauksessa PLC-logiikan muuttujista.

Asset-, aspect- ja type-profiilien luonnin jälkeen suoritettiin Onboarding-prosessi MindSpheren käyttöönottoa varten MindConnect-laitteelle. Tämä tarkoittaa, että MindConnect Nano -laite alustettiin luodun verkkoprofiilin perusteella muistitikulle syötetyn

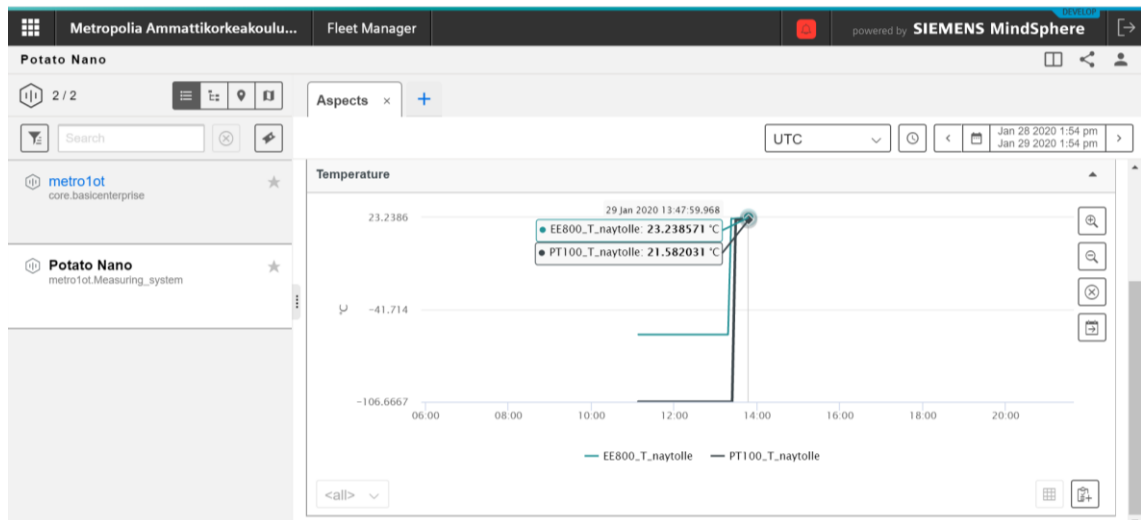
tiedoston välityksellä. Syötetyt verkkoasetukset pitävät sisällään muun muassa PLC-logiikan IP-osoitteen. Alustuksen jälkeen yhteys MindSpheren ja MindConnect-laitteen välillä oli otettu käyttöön.

Yhteyden muodostuksen jälkeen määritettiin yhteys PLC-logiikan muuttujien välille sekä MindSphere-sovelluksen data point -yhteyspisteiden välille. Määrittämisen jälkeen yhteys muodostui. Kuvassa 17 on näkymä MindSphere-sovelluksen Asset Manager -ohjelmasta, jossa yhteyden luonnin jälkeen MindSphere-sovelluksessa olevat muuttujat näyttävät reaaliaikaista tietoa PLC-logiikan muuttujista. Tämän opinnäyteprojektin yhteydessä tehtiin myös ohjeet MindSphere-sovelluksen käyttöönottoa varten SIMATIC S7-1500 PLC -logiikalle. Ohjeet löytyvät liitteestä 1, jotka kertovat yksityiskohtaisemmin sovelluksen käsitteistä (liite 1, s. 37) ja PLC-logiikan muuttujien siirrosta sovellukseen (liite 1, s. 38–67).

Name	Aspect	Category	Status															
Carbon_dioxide	metroTot.Carbon_dioxide	Dynamic	Last updated 2020-01-29 15:29:00															
<table border="1"> <thead> <tr> <th>Name</th> <th>Data type</th> <th>Unit</th> <th>Max. length</th> <th>Latest value</th> </tr> </thead> <tbody> <tr> <td>EE800_CO_naytolle</td> <td>DOUBLE</td> <td>ppm</td> <td>-</td> <td>456.95892 -0h:00m</td> </tr> </tbody> </table>				Name	Data type	Unit	Max. length	Latest value	EE800_CO_naytolle	DOUBLE	ppm	-	456.95892 -0h:00m					
Name	Data type	Unit	Max. length	Latest value														
EE800_CO_naytolle	DOUBLE	ppm	-	456.95892 -0h:00m														
connectivityStatus	core.connectivitystatus	Static	Online 2020-01-29 15:28:18															
firmwareStatus	core.firmwarestatus	Static	V03.04.00.00 b001 2020-01-09 13:43:00															
Proximity_IOS	metroTot.Proximity_IOS	Dynamic	Last updated 2020-01-29 15:29:00															
<table border="1"> <thead> <tr> <th>Name</th> <th>Data type</th> <th>Unit</th> <th>Max. length</th> <th>Latest value</th> </tr> </thead> <tbody> <tr> <td>Capasitive_sensor</td> <td>BOOLEAN</td> <td>-</td> <td>1</td> <td>-0h:00m</td> </tr> </tbody> </table>				Name	Data type	Unit	Max. length	Latest value	Capasitive_sensor	BOOLEAN	-	1	-0h:00m					
Name	Data type	Unit	Max. length	Latest value														
Capasitive_sensor	BOOLEAN	-	1	-0h:00m														
status	core.agentstatus	Dynamic	Last updated 2020-01-29 15:29:00															
Temperature	metroTot.Temperature	Dynamic	Last updated 2020-01-29 15:29:00															
<table border="1"> <thead> <tr> <th>Name</th> <th>Data type</th> <th>Unit</th> <th>Max. length</th> <th>Latest value</th> </tr> </thead> <tbody> <tr> <td>EE800_T_naytolle</td> <td>DOUBLE</td> <td>°C</td> <td>-</td> <td>22.437428 -0h:00m</td> </tr> <tr> <td>PT100_T_naytolle</td> <td>DOUBLE</td> <td>°C</td> <td>-</td> <td>21.507162 -0h:00m</td> </tr> </tbody> </table>				Name	Data type	Unit	Max. length	Latest value	EE800_T_naytolle	DOUBLE	°C	-	22.437428 -0h:00m	PT100_T_naytolle	DOUBLE	°C	-	21.507162 -0h:00m
Name	Data type	Unit	Max. length	Latest value														
EE800_T_naytolle	DOUBLE	°C	-	22.437428 -0h:00m														
PT100_T_naytolle	DOUBLE	°C	-	21.507162 -0h:00m														

Kuva 17. MindSphere-näkymä, jossa luodut muuttujat näyttävät "Latest value"-kohdassa PLC-logiikan muuttujien mukaista arvoa.

PLC-logiikan muuttujien ja MindSpheren-sovellukseen tehtyjen muuttujien välinen yhteys mahdollisti anturien antaman datan tarkastelun MindSphere-sovelluksen Fleet Manager -palvelun kautta, jossa toiminto tulosti reaaliaikaista käyrää muuttujien antamasta tiedosta. Kuvassa 18 on nähtävissä Fleet Manager -ohjelman ikkuna, jossa kahden lämpötilamuuttujan antamat arvot näkyvät aikajaksoisena käyränä.



Kuva 18. Fleet Manager -ikkuna, jossa kahden lämpötilamuuttujana antama data aikajaksoisena käyränä.

5.6.6 Datan tuonti MindSphere-sovelluksesta ja sen siistiminen

Fleet Manager -ohjelmalla tulostettiin muuttujien kautta kulkema data halutulta aikaväliltä .csv-tiedostomuodossa. Tiedostoissa oli tietyltä aikaväliltä tiedot kahden lämpötila-anturin ja hiilidioksidianturin, eli PT100-lämpötila-anturin, EE800-lämpötila-anturin ja EE800-hiilidioksidianturin, omille muuttujilleen syöttämistä datasta.

MindSphere-sovellus luki dataa PLC-logiikalta kerran sekunnissa. Datan keräyksen aikaväli oli vajaan kahden tunnin pituinen, joten dataa kerääntyi yli 6000 riviä. Dataa piti siivota, koska datasta haluttiin sekä opetusdatajoukko että testidatajoukko. Datan siivouksessa käytettiin Microsoft Excel -ohjelmaa, jonka avulla .csv-tiedostomuotoista tiedostoa pystyi käsittelemään. Taulukossa 4 näkyy MindSphere-sovelluksen Fleet Manager -ohjelmasta tuotu lämpötilamuuttujien datasta muutama rivi siivoamattomana.

Taulukko 4. Lämpötilamuuttujien antama data siivoamattomana.

EE800_T_naytolle,EE800_T_naytolle_qc,PT100_T_naytolle,PT100_T_naytolle_qc,time
-59.25926,0,-106.666664,0,"2020-01-29T13:22:41.955Z"
-59.25926,0,-106.666664,0,"2020-01-29T13:22:42.957Z"
22.428387,0,-106.666664,0,"2020-01-29T13:22:43.959Z"
22.428387,0,-106.666664,0,"2020-01-29T13:22:44.961Z"

Kuten taulukko 4 osoittaa, datassa on muokattavaa. Microsoft Excelin toimintojen avulla datan pystyi erottelemaan erikseen omiin soluihinsa. Datatiedostot yhdistettiin ja erotettiin omiksi datajoukkotiedostoiksi. Lisäksi osa datasta jouduttiin poistamaan niiden puuttumisen takia, jolloin koko rivi poistettiin. Ajasta poistettiin päivämäärä sen irrelevanttisuuden takia datan ottoa kohtaan. Näiden lisäksi ajan vierelle lisättiin sekuntimäärä Python-ohjelman käyttöä varten. Taulukossa 5 näkyy siivotun testausdatajoukon muutama rivi.

Taulukko 5. Testausdatajoukko siivottuna.

temp1	temp2	co2	time	seconds
20.572918	23.920355	389.53992	13:38:56	49136
20.570023	23.922165	389.53992	13:39:01	49141
20.564236	23.925781	397.64178	13:39:06	49146
20.57581	23.922165	397.7141	13:39:11	49151

5.6.7 Koneoppimissovelluksen luominen

Koneoppimissovelluksen tekoon käytettiin Python-ohjelmointikieltä ja Anaconda-jakelualustaa. Vaihtoehtoja koneoppimissovelluksen tekoon olisi löytynyt muitakin vaihtoehtoja, kuten Googlen Colaboratory-palvelu, mutta projektissa päädyttiin Anaconda-jakelualustaan. Anaconda-jakelualusta piti sisällään oman Python-tulkin ja alusta tarjosi tarvittavat kirjastot koneoppimista varten.

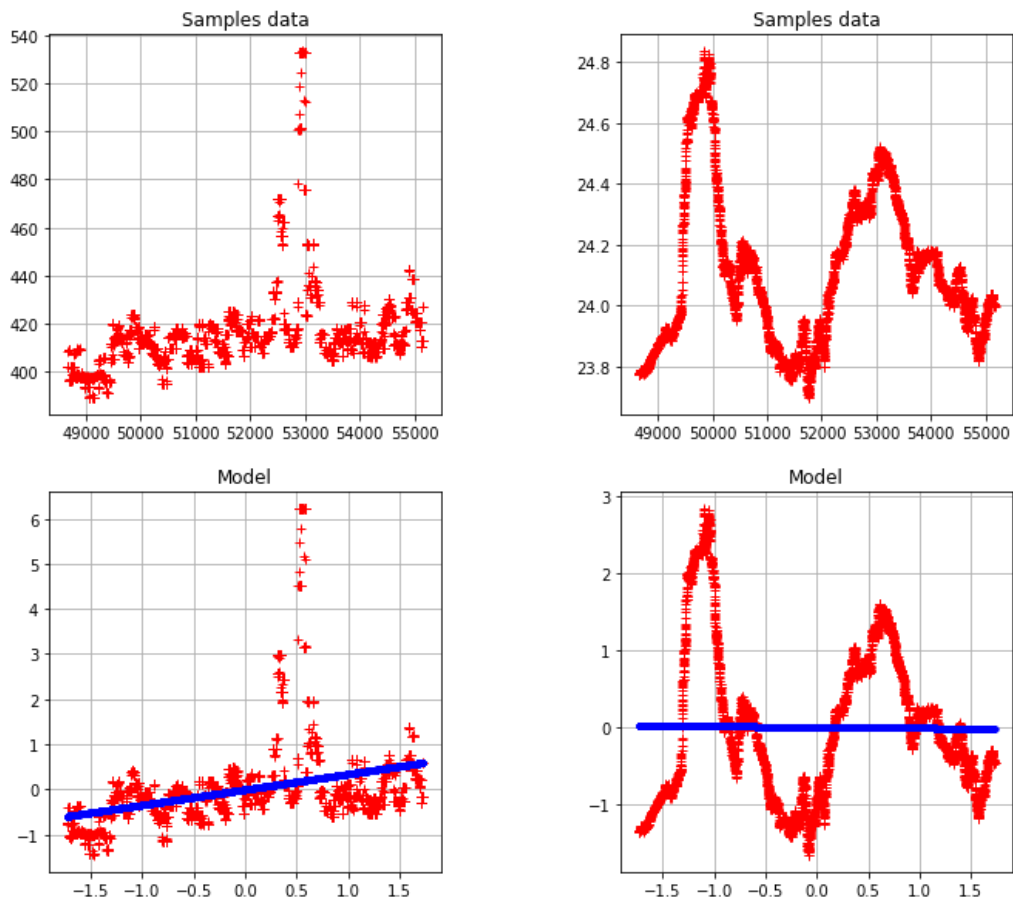
Koneoppimissovelluksen rakennuksessa päädyttiin MindSphere-sovelluksesta kerätyn datan perusteella lineaariregressiosovelluksen tekoon. Anaconda-jakelualusta asennettiin projektin käytössä olleelle tietokoneelle. Anaconda-jakelualustan mukana tullutta Jupyter Notebook -sovellusta käytettiin koneoppimissovelluksen tekoon. Koneoppimissovellukseen käytetyn Python-ohjelmakielen pohja löytyi Altoroksen Internet-sivuilta, joka on IT-alan yrityksenä erikoistunut pilvipalveluiden ja tekoälyn konsultointiin [57; 58.] Python-koodia muokattiin erityisesti muuttujien, kommentoinnin ja ohjelmaan tuodun datan osalta. Ohjelmakoodit tehtiin jokaiselle MindSphere-sovelluksesta tuodulle muuttujalle erikseen.

6 Tulokset ja analysointi

6.1 Tulokset

Työn tavoitteena oli saada aikaiseksi koneoppimissovellus, joka parantaisi aeroponisen perunankasvatuksen prosessia. Tulokseksi projektissa saatiin aikaiseksi lineaariregressiomalli, jonka esimerkki on nähtävissä kuvassa 19. Mallin luontiin käytettiin projektin aikana PLC-logiikan, anturien ja IoT-rajapintasovelluksen välityksellä kulkenutta dataa. Mallia kokeiltiin kaikille kolmelle analogista dataa välittäneelle muuttujalle, jotka olivat kahden lämpötilan ja yhden hiilidioksiditason jakavat muuttujat. Näistä kahta käytetään projektin lopputuloksena ja onnistuneena mallin muodostajana.

Työn ohessa saatiin aikaan tulokseksi myös MindSphere-sovelluksen käyttöönoton ohje, joka on nähtävissä liitteessä 1. Ohje kertoo MindSphere-sovelluksen käyttöönotosta, PLC-logiikan muuttujien siirtämisestä sovellukseen ja datan tarkastelusta.



Kuva 19. Lineaariregressiomalli.

Kuvassa 19 on nähtävissä algoritmien antamat kuvaajat. Kuvan ensimmäinen kuvaaja vasemmassa yläkulmassa on hiilidioksidiarvojen opetusjoukon datapisteet. Toinen kuva oikeassa ylälaidassa on lämpötila-arvon opetusjoukon datapisteet. Vasemmassa alalaidassa on standardoitu kuvaaja hiilidioksidiarvojen opetusjoukon datapisteistä ja lisäksi kuvaajassa oleva sininen suora on opetusjoukon, testijoukon ja tehtävässä olevin laskutoimitusten perusteella arvoihin sovitettu suora. Oikeassa alalaidassa on samanlainen standardoitu kuvaaja ja sovitettu suora lämpötila-arvolle.

Kuvassa 20 on nähtävissä tilastotietoa Python-koodin muodostaman mallin iterointi-kierroksista kustannusfunktion, lineaarisen suoran kulmakertoimen (W) ja y-akselin koordi-

naattien suoran sijainnin (b) osalta. Tulos on hiilidioksididatan perusteella tehdystä mallista. Kuvan alalaidassa toiseksi alin rivi kertoo opetuksen jälkeiset tilastot kustannusfunktion tuloksesta ja lineaarisen suoran arvioiduista muuttujista. Alin rivi kertoo testausdatajoukon kustannusfunktion tuloksesta.

```
Iteration: 0001 cost= 1.005001783 W= 0.055828735 b= -1.0222851
Iteration: 0021 cost= 0.449865341 W= 0.30716008 b= -0.124286
Iteration: 0041 cost= 0.441659808 W= 0.33771613 b= -0.015110275
Iteration: 0061 cost= 0.441538692 W= 0.34143105 b= -0.0018370599
Iteration: 0081 cost= 0.441536665 W= 0.3418827 b= -0.00022335164
Iteration: 0101 cost= 0.441536576 W= 0.34193757 b= -2.715773e-05
Iteration: 0121 cost= 0.441536695 W= 0.34194425 b= -3.3017166e-06
Iteration: 0141 cost= 0.441536665 W= 0.34194508 b= -4.045581e-07
Iteration: 0161 cost= 0.441536695 W= 0.34194508 b= -5.5125895e-08
Iteration: 0181 cost= 0.441536754 W= 0.34194508 b= -1.4147697e-08
Tuning completed: cost= 0.441536754 W= 0.34194508 b= -1.4147697e-08
Testing data cost: 0.11038923
```

Kuva 20. Python-koodilla luodun lineaariregressioalgoritmin antama tulos mallin toiminnasta hiilidioksidimuuttujan osalta.

Kuvassa 21 nähdään lämpötilamuuttujan datasta tehtyä tilastotietoa mallin muodostuksen iterointi-kierroksista. Mallissa toistetaan samat iteroinnit kuin hiilidioksidimuuttuja osalta.

```
Iteration: 0001 cost= 0.716477096 W= -0.6127421 b= -0.2674263
Iteration: 0021 cost= 0.503135204 W= -0.08454156 b= -0.03251282
Iteration: 0041 cost= 0.499981791 W= -0.02032471 b= -0.0039527966
Iteration: 0061 cost= 0.499935418 W= -0.012517441 b= -0.0004805639
Iteration: 0081 cost= 0.499934524 W= -0.011568252 b= -5.8415488e-05
Iteration: 0101 cost= 0.499934524 W= -0.011452853 b= -7.0943943e-06
Iteration: 0121 cost= 0.499934435 W= -0.011438817 b= -8.530426e-07
Iteration: 0141 cost= 0.499934435 W= -0.011437089 b= -1.04259186e-07
Iteration: 0161 cost= 0.499934524 W= -0.011436874 b= -2.18624e-09
Iteration: 0181 cost= 0.499934524 W= -0.011436844 b= -5.1664726e-09
Tuning completed: cost= 0.499934524 W= -0.011436844 b= -5.1664726e-09
Testing data cost: 0.12493329
```

Kuva 21. Lineaariregressioalgoritmissa tapahtuvan iteroinnin tulokset lämpötilamuuttujan osalta.

Tuloksista jäivät puuttumaan PLC-logiikassa mukana olleen ja kytketyn pinnankorkeusanturin tuottama data, sillä digitaaliselle totuusarvoviestille ei kehittynyt tarvittavaa käyttöä projektin osalta eikä sopivaa algoritmia datalle löytynyt. Lisäksi kolmas analoginen anturi, joka mittasi toista lämpötila-arvoa, jätettiin pois tuloksista arvojen suuren heitelyn vuoksi. Tämä olisi vääristänyt algoritmin muodostamaa mallia ylisovituksella.

6.2 Analyysi

Lineaariregression avulla saadut kuvaajat datapisteistä kertovat PLC-logiikalla ja MindSphere-ohjelmalla suodatetun datan heittelystä. Datapisteitä on sekunnin välein, joten kuvaaja on melko yhtenäinen alusta loppuun. Dataa kerättiin lähes kahden tunnin ajan, joten datapisteitä kerääntyi lähes 7000 riviä, joista osa karsiintui vikaantuneen datan takia. Datapisteet jaettiin opetusjoukkoon, joka sisälsi noin 5000 riviä dataa ja testijoukkoon, joka sisälsi noin 1300 riviä dataa. Koeympäristö datankeräyksen aikaan oli sisätila, jossa olevien ihmisten määrä oli vakio, joten datassa olevat heittelyt lämpötilan mittauksessa johtuvat anturien mahdollisesta epätarkkuudesta sekä lämpötilojen pienistä vaihteluista. Hiilidioksidia mitannut anturin yksi piikki johtuu hiilidioksiditason äkillisestä muutoksesta. Jatkossa datapisteiden keräyksen ei tarvitse olla välttämättä yhtä yhtenäinen, jotta kuvaajaa näyttäisi luettavammalta. Mallin muodostuksen kannalta sillä ei ole väliä, mutta datapisteiden vähentäminen luo ominaisuudet mallin ennustavuudelle.

Datajoukkoja luotiin kaksi, joista toista käytettiin opetusjoukkona ja toista testijoukkona. Kuvasta 20 voidaan todeta, että testijoukon kustannusfunktio on pienempi kuin opetusjoukon vastaava. Täten voidaan todeta, että malli on tarkempi testijoukolla, jonka otoksia on vähemmän kuin opetusjoukolla. Kuten kuvassa 19 voidaan huomata, lämpötilan opetusjoukossa on suuria heilahteluja, mutta lämpötilan iteroinneissa kuvassa 21, malli tarkentuu testijoukon osalta.

Algoritmissa muodostettu malli on tehty datajoukkojen ja tehtävässä käytettyjen laskutoimitusten perusteella. Suora osoittaa datapisteiden erojen muodostamaa sovitetta, joka on ennuste datapisteiden perusteella tehdystä arviosta datan käyttäytymisestä aikavälillä. Ennusteeseen vaikuttavat standardoitu aikamuuttuja sekä lineaarisen suoran yhtälö, jossa käytetään käytännöllisintä suoran kulmakerrointa ja suoran sijaintia y-akselilla. Painomuuttuja eli suoran kulmakerroin ja bias-muuttuja eli suoran sijainnin y-akselilla määrittävä muuttuja saadaan iteroimalla x- ja y-akselin dataa tarpeeksi monta kertaa kustannusfunktion ja gradient descent -algoritmin läpi. Kuvista 20 ja 21 voidaan todeta, että lineaariregression ohjelmakoodissa paino- ja bias-muuttujat löydetään iterointien alkuvaiheessa.

Lineaariregressiomallia voitaisiin käyttää opettamalla mallia pitemmällä aikavälillä, kuten esimerkiksi yhden perunasadon kasvun ajalta. Tästä opetuksesta muodostunutta mallia voitaisiin käyttää ennustuksiin sadon käyttäytymisestä kasvuaikana. Mallin ennustuksiin nojautuen tietoa voitaisiin hyödyntää esimerkiksi ravintoliuoksen ohjauksessa, yksittäisten kasvutekijöiden käyttäytymisessä sadon tietyssä vaiheessa tai sadon onnistumisen ennustuksessa. Tuloksen lukuun ei käytetty perunankasvatuksen asiantuntijaa, joten tulosten oikeellisuutta ei voida taata. Tuloksen lukuun on käytetty yksinomaan mallin muodostukseen käytettyä ja koneoppimismallin tulostamaa dataa.

Projektissa käytettiin ympäristön seurantaan kahta ympäristössä olevaa tekijää anturien mittauksessa, jotka olivat hiilidioksidi ja lämpötila. Projektin puutteeksi jäivät muiden aëronomisen perunankasvatusprosessin anturien hankkiminen, jotka olisivat tuottaneet lisätietoa ympäristöstä ja prosessista. Erityisesti tieto sähkönjohtokyvystä ja pH-tasosta vaikuttavat kasvattajan toimiin kasvuprosessin aikana, mitkä auttavat kasvua oikeaan suuntaan. Projektissa käytettiin hiilidioksidi- ja lämpötila-antureita, joiden antamalla datalla oli mahdollista muodostaa koneoppimisalgoritmi. Algoritmin muodostuksen kannalta data on tärkeää ja tulevaisuuden kannalta mitä useammasta lähteestä dataa saa muodostettua, sen tarkemmaksi ja monikäyttöisemmäksi malli muotoutuu. Useammassa datan lähteessä on kuitenkin riskinsä mallin muodostuksen ja ylisovituksen kannalta.

Projektin lopputuloksena oli siis muodostettu koneoppimismalli, ja seuraavaksi edessä olisi tämän mallin tutkiminen hyväksyttävänä metodina perunankasvatusprosessissa. Mallia voisi käyttää useamman sadon kasvatuksessa anturien kerätessä tietoa ympäristön lämpötilan ja hiilidioksidin vaihtelusta. Mallin käytön ohella tutkittaisiin prosessissa tapahtuvia muutoksia ja korreloitaisiin anturien antaman datan ja prosessin muutoksien välistä yhteyttä. Havainnoilla voitaisiin kehittää mallin käyttöä perunankasvatuksessa. Mallin käytön haittapuolina on sen sopeutuvuus ympäristön muuttujiin. Äkilliset hiilidioksidi- ja lämpötilatason heittelyt voivat sopeuttaa mallin väärin, mikä vaikuttaa tuleviin ennustuksiin ja niiden käytettävyyteen.

Lisäksi projektin jälkeen tutkintakohteena on koneoppimisalgoritmin tuottaman datan siirto takaisin PLC-logiikalle. Algoritmin avulla olisi mahdollista vaikuttaa esimerkiksi PLC-logiikan välityksellä tehtäviin ravintoliuoksen ohjauksiin. Koneoppimisen ohjelmointikielenä toimi tässä projektissa Python, kun taas PLC-logiikan yhteydet sisältävät joukon

erilaisia protokollia ja Siemensin logiikoiden ohjelmoinnissa avustava TIA Portal -ohjelmisto mahdollistaa neljän erilaisen logiikkaohjelmointikielen käytön. Yksi mahdollinen datan siirto tapa olisi Python-kielen kirjasto Snap7, joka mahdollistaa yhteydenpidon Python-koodin ja Siemensin S7 PLC -logiikoiden välille [59].

7 Yhteenveto

Opinnäytetyöprojektissä tutustuttiin tekoälyn ja koneoppimisen soveltamiseen automatisoidun aeroponisen perunankasvatusjärjestelmän yhteydessä. Työn tarkoituksena oli tutustua tekoälyn ja koneoppimisen käyttöön automatisoinnin kannalta ja samalla parantaa aeroponista perunankasvatusprosessia. Projektissa tutkimuksen kohteena oli aeroponinen perunankasvatusjärjestelmä. Järjestelmä yleisesti koostuu kasvatussäiliöstä, jossa roikkuvat istutetut ilmaperunat, sekä kastelujärjestelmästä, jonka avulla perunoiden juurille tihkutetaan ravintoliuosta. Automatisointi koostui perunankasvatusprosessissa käytettävien anturien, Siemensin PLC-logiikan ja PLC-logiikan tiedonsiirron IoT-rajapintasovellukseen soveltuvan MindConnect-laitteen hankinnasta ja niiden kytkemisestä toisiinsa. Lopuksi IoT-rajapintasovellus MindSpheren avulla saadusta datasta kehitettiin koneoppimisalgoritmi, jossa muodostettiin lineaariregressiomalli.

Projektin aikana tehdyt vaiheet olivat kaikki projektin suorittajalla uusia vaiheita lukuun ottamatta PLC-logiikan ohjelmointia, joten niin hankinnat, anturien kytkennät PLC-logiikkaan ja IoT-rajapintasovelluksen käyttöönotto olivat opettavaisia vaiheita. Tekoäly ja koneoppiminen, jotka olivat projektin pääteeman osat, olivat ennestään tuttuja teorian osalta, mutta projektin aikana tätä tietoutta pääsi syventämään. Lisäksi projektin aikana koneoppimisen käyttöönotto ja käytäntö tulivat tutuiksi. Projektin jälkeen tuntuukin, että koneoppimisesta on vasta pintaraapaisu otettu, sillä aiheen haarat ovat laajat. Projektin aikana myös Python-ohjelmointikieli tuli tutuksi ja projektin aikana käytetty aika kielen opetteluun (40 h) auttoi sisäistämään kielen käytön tapoja ja muun muassa tiedoston tuonnin Python-kielillä muodostettuun ohjelmaan.

Projekti oli osa Digi-Salama-hanketta ja yksi hankkeen tuloksista. Projektissa tulokseksi saatu lineaariregressiomalli muodostettiin käytössä olleiden lämpötila- ja hiilidioksidian-

turin antaman datan perusteella. Lisäksi mallin muodostuksessa apuna oli lineaari-regressiomallin muodostuksen keinot, kuten lineaarisen suoran yhtälö. Tavoitteet saavutettiin konkreettisen mallin muodossa. Koneoppimisen ala on kuitenkin laaja, jonka metodeihin tutustutaankin tässä opinnäytetyössä. Laajuuden takia algoritmin löytäminen muodostui hankalaksi. Lisäksi datan liittäminen malliin osoittautui vaikeaksi tehtäväksi, koska jokainen algoritmi tarvitsee esitietoa omasta reaali maailman prosessistaan. Kaiken kaikkiaan koneoppimissovelluksen muodostaminen ei ole pelkkää koodin kirjoitusta, vaan tarvitaan paljon taustatietoa kehitettävästä prosessista, koska mallin muodostuksen jälkeen käyttäjien on osattava lukea sitä.

Vaikka projekti päättyikin lineaariregressiomalliin, niin seuraavaksi edessä olisi mallin käyttäminen datan keräykseen pitemmältä aikaväliltä. Datan keräyksellä olisi mahdollista kehittää mallia paremmaksi ja käyttää sitä hyödyksi tulevien perunasatojen kasvatuksessa. Lisäksi tutkittavaksi jää metodi, jonka avulla koneoppimissovelluksen sisältämä data saataisiin ajettua takaisin PLC-logiikalle, joka pystyisi toimintoissaan hyödyntämään tätä dataa.

Lähteet

- 1 Hanke. 2020. Verkkoaineisto. Digi-Salama. <<https://digisalama.metropolia.fi/hanke/>>. 24.3.2020. Luettu 9.5.2020.
- 2 Urban Farm Lab. 2020. Verkkoaineisto. Sitra. <<https://www.sitra.fi/hankkeet/urban-farm-lab/>>. 24.4.2020. Luettu 9.5.2020.
- 3 Mbiyu, M.W.; Muthoni, J.; Kabira, J.; Elmar, G.; Muchira, C.; Pwaipwai, P.; Ngauriia, J.; Otieno, S.; Onditi, J. 2012. Use of aeroponics technique for potato (*Solanus tuberosum*) minitubers production in Kenya. Verkkoaineisto. <<https://cgspace.cgiar.org/bitstream/handle/10568/66549/76745.pdf?sequence=1&isAllowed=y>>. 6.11.2012. Luettu 3.3.2020.
- 4 Clark, Josh. 2008. How Aeroponics Works. Verkkoaineisto. HowStuffWorks. <<https://home.howstuffworks.com/lawn-garden/professional-landscaping/aeroponics1.htm>>. 22.12.2008. Luettu 4.3.2020.
- 5 Lumme, Marika. 2019 Keskellä Myyrmäkeä on ainutlaatuinen paikka, jossa perunat kasvavat ilmassa – antaa jopa 10-kertaisen sadon tavanomaiseen verrattuna. Verkkoaineisto. Vantaan Sanomat. <<https://www.vantaansanomat.fi/paikalliset/1238677>>. 15.5.2019. Luettu 4.3.2020.
- 6 Lakhiar, Imran Ali; Gao, Jianmin; Syed, Tabinda Naz; Chandio, Farman Ali; Buttar, Noman Ali; Qureshi, Waqar Ahmed. 2018. Monitoring and Control Systems in Agriculture Using Intelligent Sensor Techniques: A Review of the Aeroponic System. PDF-dokumentti. Journal of Sensors. 19.12.2018. Luettu 24.3.2020.
- 7 Lakhiar, Imran Ali; Gao, Jianmin; Syed, Tabinda Naz; Chandio, Farman Ali; Buttar, Noman Ali. 2018. Modern plant cultivation technologies in agriculture under controlled environment: a review on aeroponics. PDF-dokumentti. Journal of Plant Interactions. 30.5.2018. Luettu 15.4.2020.
- 8 Ahmadi Lahijani, M.J.; Kafi, M.; Nezami, A.; Nabati, J.; Erwin, J.E. 2019. Effect of CO₂ Enrichment on Gas Exchanges, Biochemical Traits and Minituber Yield in Potato (*Solanum tuberosum* L.) Cultivars. PDF-dokumentti. Journal of Agricultural Science and Technology Volume 21. 24.6.2019. Luettu 2.5.2020.
- 9 Difference between Artificial intelligence and Machine learning. 2019. Verkkoaineisto. Javatpoint. <<https://www.javatpoint.com/difference-between-artificial-intelligence-and-machine-learning>>. 28.7.2019. Luettu 24.2.2020.

- 10 Reese, Hope. 2017. Understanding the differences between AI, machine learning, and deep learning. Verkkoaineisto. TechRepublic. <<https://www.techrepublic.com/article/understanding-the-differences-between-ai-machine-learning-and-deep-learning/>>. 23.2.2017. Luettu 4.5.2020.
- 11 What is Artificial Intelligence? 2018. Verkkoaineisto. Built In. <<https://builtin.com/artificial-intelligence>>. 14.9.2018. Luettu 11.3.2020.
- 12 Mueller, John Paul; Massaron, Luca. 2018. Artificial Intelligence for dummies. Hoboken, New Jersey: John Wiley & Sons, Inc.
- 13 Ray, Shaan. 2018. History of AI. Verkkoaineisto. Towards Data Science. <<https://towardsdatascience.com/history-of-ai-484a86fc16ef>>. 11.8.2018. Luettu 25.3.2020.
- 14 VijiPriya, Dr J.; Ashok, Dr Jammi; Suppiah, Dr S. 2016. A Review on Significance of Sub Fields in Artificial Intelligence. Verkkodokumentti. International Journal of Latest Trends in Engineering and Technology Volume 6 Issue 3 January 2016. <<https://www.ijltet.org/journal/86.pdf>>. 31.1.2016. Luettu 15.4.2020.
- 15 Flach, Peter. 2012. Machine Learning The Art and Science of Algorithms that Make Sense of Data. University Printing House, Cambridge, UK.
- 16 Algorithm. 2020. Verkkoaineisto. Merriam-Webster Dictionary. <<https://www.merriam-webster.com/dictionary/algorithm>>. Luettu 19.4.2020.
- 17 Raschka, Sebastian; Mirjalili, Vahid. 2017. Python Machine Learning. E-kirja. Packt Publishing.
- 18 Shah, Tarang. 2017. About Train, Validation and Test Sets in Machine Learning. Verkkoaineisto. Towards Data Science. <<https://towardsdatascience.com/train-validation-and-test-sets-72cb40cba9e7>>. 6.12.2017. Luettu 16.4.2020.
- 19 Brownlee, Jason. 2016. Overfitting and Underfitting With Machine Learning Algorithms. Verkkoaineisto. Machine Learning Mastery. <<https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/>>. 21.3.2016. Luettu 16.4.2020.
- 20 Brownlee, Jason. 2016. Supervised and Unsupervised Machine Learning Algorithms. Verkkoaineisto. Machine Learning Mastery. <<https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/>>. 16.3.2016. Luettu 18.4.2020.

- 21 Brownlee, Jason. 2020. 4 Types of Classification Tasks in Machine Learning. Verkkoaineisto. Machine Learning Mastery. <<https://machinelearningmastery.com/types-of-classification-in-machine-learning/>>. 8.4.2020. Luettu 17.4.2020.
- 22 Brownlee, Jason. 2017. Difference Between Classification and Regression in Machine Learning. Verkkoaineisto. Machine Learning Mastery. <<https://machinelearningmastery.com/classification-versus-regression-in-machine-learning/>>. 11.12.2017. Luettu 17.4.2020.
- 23 Zammataro, Luca. 2019. Linear Regression with one or more variables. Verkkoaineisto. Towards Data Science. <<https://towardsdatascience.com/machine-learning-for-biomedical-data-linear-regression-7d43461cdfa9>>. 9.12.2019. Luettu 17.4.2020.
- 24 Gandhi, Rohith. 2018. Introduction to Machine Learning Algorithms: Linear Regression. Verkkoaineisto. Towards Data Science. <<https://towardsdatascience.com/introduction-to-machine-learning-algorithms-linear-regression-14c4e325882a>>. 27.5.2018. Luettu 18.4.2020.
- 25 Pandey, Parul. 2019. Understanding the Mathematics behind Gradient Descent. Verkkoaineisto. Towards Data Science. <<https://towardsdatascience.com/understanding-the-mathematics-behind-gradient-descent-dde5dc9be06e>>. 18.3.2019. Luettu 24.4.2020.
- 26 Sammut, Claude; Webb, Geoffrey I. 2011. Encyclopedia of Machine Learning. E-kirja. Springer Publishing Company.
- 27 PT100 Sensor Working Principle. 2019. Verkkoaineisto. Process parameters Ltd. <<https://www.processparameters.co.uk/pt100-sensor-working-principle/>>. 25.6.2019. Luettu 4.3.2020.
- 28 Jumo dTrans T03 J, B, T Analog 2-wire transmitter with digital adjustment. 2002. PDF-dokumentti. Jumo. 27.2.2002. Luettu 20.4.2020.
- 29 HVAC Room Transmitter for CO2, Temperature and Relative Humidity. 2016. Verkkoaineisto. E+E Elektronik. <https://www.elfadistelec.fi/Web/Downloads/_t/ds/EE800_eng_tds.pdf?pid=30069729>. 8.7.2016. Luettu 9.3.2020.
- 30 NDIR explained. 2015. Verkkoaineisto. Edaphic Scientific. <<https://www.edaphic.com.au/knowledge-base/articles/gas-articles/ndir-explained/>>. 14.6.2015. Luettu 9.3.2020.
- 31 Flat Type Capacitive Proximity Sensor E2K-F. 2005. Verkkoaineisto. RS Components Ltd. <<https://docs.rs-online.com/790b/0900766b812f7a04.pdf>>. 10.8.2005. Luettu 1.5.2020.

- 32 SIMATIC ET 200SP Digital input module DI 8x24 VDC HF Edition 12/2017. 2017. PDF-dokumentti. Siemens. 6.12.2017. Luettu 1.4.2020.
- 33 MJ3055T Complementary Silicon Power Transistors. 2003. Verkkoaineisto. ST. <<https://www.st.com/resource/en/datasheet/mje3055t.pdf>>. 30.9.2003. Luettu 20.4.2020.
- 34 CPU 1512SP-1 PN. 2017. Verkkoaineisto. Siemens. <<https://mall.industry.siemens.com/mall/en/WW/Catalog/Products/10239949?activeTab=productinformation®ionUrl=WW>>. 28.11.2017. Luettu 10.3.2020.
- 35 Wilson, Justin. 2018. What is the difference between PROFIBUS and PROFINET? Verkkoaineisto. RealPars. <<https://realpars.com/difference-between-profibus-and-profinet/>>. 19.11.2018. Luettu 10.3.2020.
- 36 SIMATIC ET 200SP BaseUnits Edition 02/2018. 2018. Verkkoaineisto. Siemens. <http://media.automation24.com/manual/se/59753521_et200sp_base_units_manual_en-US_en-US.pdf>. 22.2.2018. Luettu 10.3.2020.
- 37 SIMATIC ET 200SP Server module Manual Edition 03/2015. 2015. PDF-dokumentti. Siemens. 24.3.2015. Luettu 1.4.2020.
- 38 SIMATIC ET 200SP Analog input module AI 4xU/I 2-wire ST Edition 09/2018. 2018. PDF-dokumentti. Siemens. 27.9.2018. Luettu 1.4.2020.
- 39 Mäenpää, Janne. 2015. TIA Portal, S7-1200 ja WinCC -käyttöönotto. Insinöörityö. Metropolia Ammattikorkeakoulu. Theseus-tietokanta.
- 40 Scaling of STEP 7 and WinCC in the TIA Portal. 2015. Sovelluksen ohjedokumentti. Siemens. 1.12.2015. Luettu 1.4.2020.
- 41 Berger, Hans. 2016. Automating with SIMATIC: Hardware and Software, Configuration and Programming, Data Communication, Operator Control and Monitoring. E-kirja. John Wiley & Sons.
- 42 SFS-EN 61131-3. Programmable controllers - Part 3: Programming languages. 2013. Standardi. Suomen Standardisoimisliitto SFS.
- 43 Berger, Hans. 2014. Automating with SIMATIC S7-1500. Configuring, Programming and Testing with STEP 7 Professional. E-kirja. Publicis Publishing, Erlangen.

- 44 MindSphere Enabling the world's industries to drive their digital transformations. 2018. PDF-dokumentti. Siemens. 30.8.2018. Luettu 31.3.2020.
- 45 Ahonen, Tero. 2014. Mikä ihmeen PaaS? Verkkoaineisto. Cybercom. <<https://www.cybercom.com/fi/Suomi/Yritys/Blogit/Blogit/Mika-ihmeen-PaaS/>>. 27.2.2014. Luettu 31.3.2020.
- 46 MindConnect Nano Product Sheet Edition 07/2018. 2018. PDF-dokumentti. Siemens. 20.8.2018. Luettu 31.3.2020.
- 47 Getting Connected to MindSphere Edition 01/2020. 2020. PDF-dokumentti. Siemens. 7.1.2020. Luettu 28.01.2020.
- 48 Deitel, Harvey; Deitel, Paul; Deitel, Paul J. 2019. Python for Programmers, First Edition. E-kirja. Prentice Hall.
- 49 Kasurinen, Timo. 2016. Olio-ohjelmoinnin perusteita. Kurssimateriaali. Päivitetty 13.3.2016. Luettu 21.4.2020.
- 50 Tensorflow. 2015 Verkkoaineisto. GitHub. <<https://github.com/tensorflow/tensorflow>>. 9.11.2015. Luettu 21.4.2020.
- 51 TensorFlow White Papers. 2020. Verkkoaineisto. TensorFlow. <<https://www.tensorflow.org/about/bib>>. 31.3.2020. Luettu 21.4.2020.
- 52 Individual Edition. 2020. Verkkoaineisto. Anaconda. <<https://www.anaconda.com/products/individual>>. 19.4.2020. Luettu 21.4.2020.
- 53 Anaconda Individual Edition. 2017. Verkkoaineisto. Anaconda. <<https://docs.anaconda.com/anaconda/>>. 11.9.2017. Luettu 21.4.2020.
- 54 Osterbuhr, Tim. 2019. What is Anaconda and how does it relate to Python? Verkkoaineisto. Venture Lessons. <<https://www.venturelessons.com/what-is-anaconda/>>. 21.9.2019. Luettu 4.5.2020.
- 55 Getting Started with Navigator. 2018. Verkkoaineisto. Anaconda. <<https://docs.anaconda.com/anaconda/navigator/getting-started/>>. 15.2.2018. Luettu 21.4.2020.
- 56 Driscoll, Mike. 2019. Jupyter Notebook: An Introduction. Verkkoaineisto. Real Python. <<https://realpython.com/jupyter-notebook-introduction/>>. 28.1.2019. Luettu 22.4.2020.

- 57 Altoros Main Page. 2007. Verkkoaineisto. Altoros. <<https://www.altoros.com/>>. 13.3.2007. Luettu 22.4.2020.
- 58 Kovalev, Sergey. 2016. Implementing Linear Regression with TensorFlow. Verkkoaineisto. Altoros. <<https://www.altoros.com/blog/using-linear-regression-in-tensorflow/>>. 13.4.2016. Luettu 22.4.2020.
- 59 Python-snap7. 2013. Verkkoaineisto. GitHub. <<https://github.com/gijzelaerr/python-snap7>>. 19.10.2013. Luettu 29.4.2020.

MindSphere-sovelluksen käyttöönotto-ohjeet

Jari Ruuska

Siemensin S7-1500 logiikan muuttujien yhdistäminen MindSphereen

Ohje Siemensin avoimen pilvipohjaisen IoT-operointijärjestelmän käyttöönottoon

Metropolia Ammattikorkeakoulu

Automaatiotekniikka

Käyttöönotto-ohje

11.5.2020

Sisällys

Lyhenteet	3
1 Johdanto	4
2 Esivalmistelut	4
2.1 Sisäänkirjautuminen MindSphereen	5
2.2 MindConnect Nano	10
2.3 MindConnect IoT2040	11
2.4 Internet- ja lähiverkkojen asennukset.	12
2.4.1 IP-osoitteiden vaihto tietokoneelta	13
2.4.2 IP-osoitteiden vaihto PLC-logiikalle	17
2.4.3 Oman TIA-projektin IP-osoitteiden vaihto	23
2.5 USB-tikun alustaminen FAT32-tiedostomuotoon	27
2.6 Portin 443 tarkistus ja sen avaus	29
3 MindSpheren käyttöönotto	37
3.1 MindSphereen liittyvät käsitteet	37
3.2 Aspectin ja tyyppin luonti	38
3.2.1 Aspectin luonti	38
3.2.2 Typen luonti	44
3.3 Assetin luonti	48
3.4 MindSphereen kytkeytyminen	50
3.4.1 MindSpheren reititykset	50
3.4.2 MindConnect-laitteen Onboarding	54
3.4.3 MindConnect-laitteen alustaminen ja uuden firmwaren asennus	56
3.5 Datasourcen ja data pointien määrittäminen	58
3.6 Fleet Managerin käyttöönotto	69
3.6.1 Datan export-toiminto Fleet Managerissa	72
4 Loppusanat	74
Lähteet	75

Lyhenteet

IoT	Internet of Things -ilmaus viittaa valloillaan olevaan ajatukseen, jossa kaikki fyysiset esineet on mahdollista yhdistää Internetiin [1]. Tässä projektissa MindSphere luokitellaan IoT-alustaksi, joka määritetään yhdistämällä neljä elementtiä: anturi, pilvi, datan prosessointi ja käyttäjärajapinta [2].
PLC	Programmable Logic Controller, ohjelmitava logiikkaohjain, jonka tuloja voidaan lukea ja lähtöjä ohjata päälle logiikkaohjaimen ajatun ohjelman komentokäskyillä.
S7	Tämän ohjeen tekoon käytetyssä projektissa käytettiin Siemensin S7-1500-sarjan logiikkaohjainta, jonka S7-protokollaa käytetään logiikkaohjaimen ohjelmointiin sekä datan vaihtamiseen S7-protokollan kanssa keskustelevien laitteiden välillä [3]. Tässä projektissa S7-protokollalla keskustelivat S7-1500-sarjan logiikka ja MindConnect-laite sekä MindSphere.
TCP	Transmission Control Protocol on yksi pääprotokollista TCP/IP-verkoissa, jonka avulla tietokoneet kommunikoivat Internetin yli. TCP tarjoaa informaatiopakettien lähetyksen siinä järjestyksessä missä ne lähetettiin sekä pakettien virhetarkistuksen. [4.]
TIA Portal	Siemensin oma suunnitteluohjelmisto (Totally Integrated Automation), joka yhdistää logiikkaohjelmoinnin, käyttöliittymien teon sekä laitteiden välisen konfiguroinnin.
UDP	User Datagram Protocol toimii kuten TCP, mutta se ei vaadi virheidentarkistusta. UDP lähettää dataa jatkuvasti vastaanottajalle riippumatta siitä vastaanottaako se sitä vai ei. [4.]

1 Johdanto

MindSphere on Siemensin avoin pilvipohjainen selaimessa toimiva IoT-käyttöjärjestelmä, jonka avulla voidaan tarkastella dataa, jonka loppukäyttäjä on kerännyt laitteistaan ja järjestelmistään. Kerätyn datan avulla voidaan säädellä tuotantoa, minimoida tuotannon seisokkeja ja varmistaa sen tehokkuus. [5.] Tässä ohjeessa käsitellään PLC-pohjaisten muuttujien määrittelemisen S7-protokollalla MindSphere-järjestelmään ja niiden antaman datan tarkastelu historiakuvaajina.

Tämä ohje on tehty opinnäytetyön ohella, jossa tutustuttiin yhtenä osana Siemensin MindSphereen. PLC-logiikkana, josta muuttujat siirretään MindSphereen, on käytetty Siemensin S7-1500-sarjan logiikkaa. PLC-ohjelma, josta muuttujat siirretään logiikkaan, on toteutettu Siemensin TIA Portalilla. Tämän ohjeen käyttöön ei tarvita aikaisempaa kokemusta pilvipohjaisista IoT-käyttöjärjestelmistä, mutta ohjetta käyttävän oletetaan tehneen TIA Portalilla PLC-ohjelman, jonka muuttujia on mahdollista käsitellä MindSphereessa. Lisäksi ohjeen vaiheet on tehty käyttäen Windows 10 -käyttöjärjestelmää, joten muiden käyttöjärjestelmien osalta toimivuutta ei voida varmistaa käyttämällä tätä ohjetta.

Ohjetta käyttävän kannattaa myös tutustua Siemensin oman ohjeen uusimpaan versioon Getting Connected to MindSphere. Tämä ohje on kirjoitettu testattaessa MindSpherea tammikuussa 2020, joten lähteenä tätä ohjetta tehdessä on käytetty Siemensin ohjeen versiota 01/2020. Ohje on pyritty toteuttamaan mahdollisimman suoraviivaisesti, joten Siemensin ohje antaa vastauksen moniin heränneisiin kysymyksiin, mihin tämä ohje ei välttämättä vastaa.

2 Esivalmistelut

MindSpheren käyttöönotto vaatii sen käyttöönottovaiheiden lisäksi monta mahdollistavaa tekijää, jotta se saadaan otettua toimintaan. Tässä kappale käsittelee vaiheet, jotka käyttäjän on varmistettava tai valmistettava, jotta MindSpheren käyttöönotto on mahdollista. Lisäksi tämän kappaleen alakappaleet neuvovat yksityiskohtaisemmin, miten ratkaista mahdolliset ongelmat, mikäli ranskalaisilla viivoilla merkityt vaiheet tuottavat niitä.

- Toimiva Internet-yhteys ja uusin selainversio MindSpheren selainkäyttöön tietokoneessa tai tabletissa. Tietokoneen tai tabletin näytön resoluution on oltava vähintään 1024x768.
- Käyttäjän oikeudet MindSpheren käyttöön. Käyttäjän pitäisi saada sähköpostiin linkki, jonka kautta käyttäjän on määritettävä sähköpostilinkin kautta itselleen käyttäjätunnus (esimerkkikäyttäjällä sähköpostiosoite) ja salasana MindSpheren kirjautumiseksi. Lisäksi käyttäjälle on annettu TenantAdmin-oikeudet MindSpheren käyttöön.
- 24 VDC:n jännitteensyöttö MindConnect-laitteeseen.
- MindConnect-laitteen, joka on päällä. Tässä ohjeessa käsitellään MindConnect Nanoa ja MindConnect IoT2040-laitetta.
- Toimiva Internet-yhteys MindConnect-laitteeseen sekä toimiva tuotantoverkko, joka yhdistää MindConnect-laitteen ja laitteen, josta MindSphereen yhdistettävien muuttujien dataa kerätään. Tässä ohjeessa dataa kerätään Siemensin S7-1500-sarjan PLC-logiikalla.
- Internet-verkon ja tuotantoverkon oikeat IP-osoitteet.
- USB-tikku, joka on alustettu FAT32-tiedostomuotoon.
- Portin 443 päälläolo, koska MindConnect-laitteet lähettävät HTTPS-yhteyksiä portin 443 kautta. Tämän huomaa helposti, mikäli yhteydet eivät toimi portin 443 kautta, sillä silloin ei toimi muukaan Internet-liikenne. [6, s. 15, 37–40, 45–48.]

2.1 Sisäänkirjautuminen MindSphereen

- Kirjaudu sähköpostiisi ja avaa sähköposti, jossa on linkki Mindsphereen. Linkki on muotoa *tilin_nimi.MindSpheren_maantieteellinen_alue.mindsphere.io*. Linkki avaa kuvan 1 mukaisen näkymän. Tallenna linkki kirjanmerkkeihisi.

SIEMENS

Sign In

or [create an account](#)

Sign In has changed. If you previously signed in with a username, please use your email. [Need help?](#)

Email

Password

[Show](#)

[Forgot your password?](#)

Sign In

Kuva 1. MindSpheren kirjautumissivu.

- Mikäli kirjaudut ensimmäistä kertaa, joudut tekemään itsellesi MindSphere-tunnukset. Sähköpostiisi pitäisi tulla myös tilin luontiviesti, jonka linkkiä painamalla avautuu kuvan 2 mukainen näkymä.

- Kuvassa 2 näkyvät tunnusten luomiseen tarvittavat tiedot. Tunnusten luontivelho kysyy nimeä, sähköpostia (tähän on syötettävä sama sähköposti, johon sähköposti on tullut), tilin salasanaa, oppilaitoksen/yrityksen nimeä ja maata. Näistä sähköposti ja tilin salasana luovat sisäänkirjautumistunnukset. Lopuksi paina Create Account.
- Siemens haluaa varmistaa sähköpostin, joten omaan sähköpostiin tulee vielä varmistussähköposti. Viestissä on tilin aktivointinappi, jonka painamisen jälkeen tunnukset ovat käytössä.

Create an Account

or [sign in](#) if you already have an account

First Name

Last Name

Email

Corporate emails preferred.

Password

[Show](#)

At least 8 characters in length and contains at least 3 of 4 types of characters: lower case(a-z), upper case (A-Z), numbers (0-9), special characters (!@#\$%^&*).

Company University Other

Organization Name

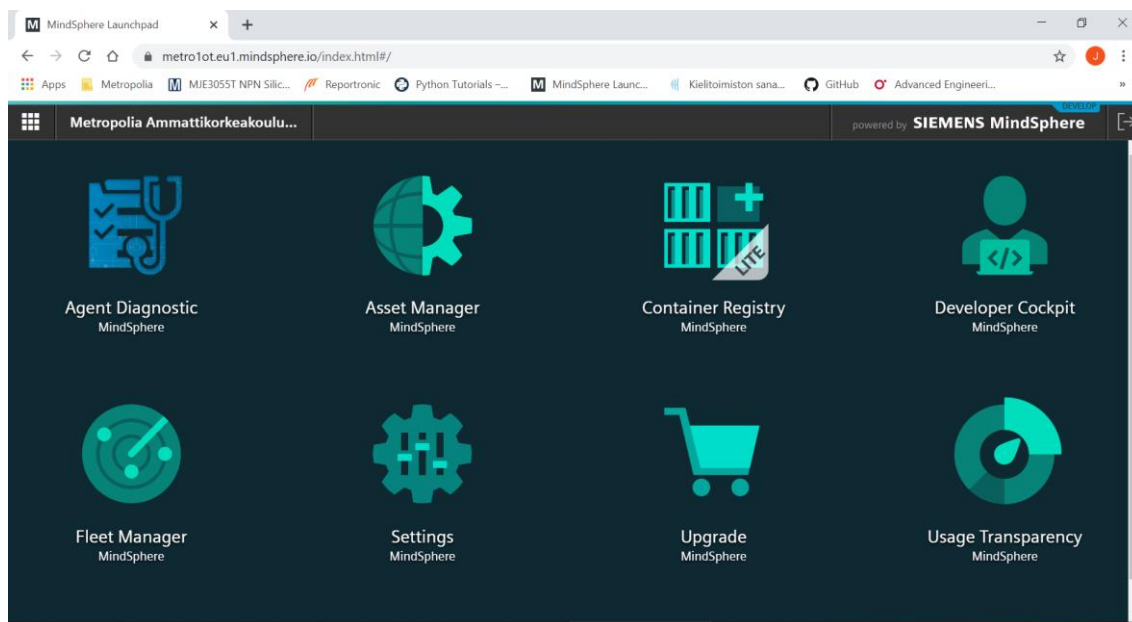
Country

Create Account

Kuva 2. Tunnusten luominen.

- Kirjaudu tunnuksillasi MindSphereen.

- Aloitusnäymän, joka näkyy kuvassa 3, Settings-painikkeesta voidaan tarkastella ja muuttaa MindSpheren asetuksia ja esimerkiksi omia oikeuksia MindSpheren käyttöön. HUOM! Mikäli tarvittavaa oikeutta TenantAdmin ei löydy käyttäjältäsi, niin käyttöönottoaiheessa tapahtuvaa aspectin, typen tai assetin luontia on mahdotonta tehdä. Pyydä silloin tilinhaltijalta käyttöoikeuksia.



Kuva 3. MindSpheren aloitusnäkö.

2.2 MindConnect Nano



Kuva 4. MindConnect Nano [6, s.18].

Kuvassa 4 oleva laite on MindConnect Nano, joka on teollisuustietokone ja mahdollistaa MindSphereen yhdistymisen. Laite päästää läpi salattua dataa, joka viedään MindSphereen turvallisen Internet-yhteyden kautta. Laite mahdollistaa koneiden ja järjestelmien IoT-yhteyden. [6, s.18.] Kuvassa 4 näkyvät neljä vihreää LED-valoa kertovat laitteen tilasta, yhteydestä MindSphereen ja mahdollisista virheistä. Getting Connected to MindSphere -ohjeen sivulta 26 löytyy ohjeet LED-valojen tiloista. Kuvasta 5 löytyy MindConnect Nano -laitteen pohja, josta löytyvät mm. 24 VDC:n jännitesyötön liitin ja virtanäppäin ympyröitynä punaisella. Lisäksi MindSphereen yhdistymisen kannalta olennaiset portit ovat X63-USB-portti konfiguraatiodiestojen latausta varten, sekä X1- ja X2- RJ45-liitinpaikat verkkoyhteyksiä varten. MindConnect Nano pystyy lukemaan 250 S7-protokollan data pointia sekunnissa [6, s.122].



- ① X63 port
- ② PN/IE LAN X1 P1 port
- ③ PN/IE LAN X2 P1 port

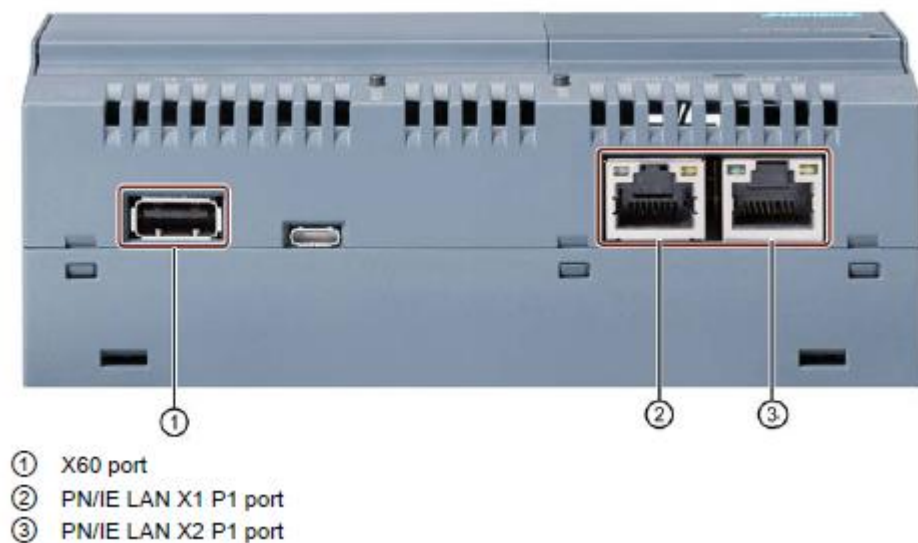
Kuva 5. MindConnect Nano -laitteen pohja [6, s.41].

2.3 MindConnect IoT2040



Kuva 6. MindConnect IoT2040 [6, s.19].

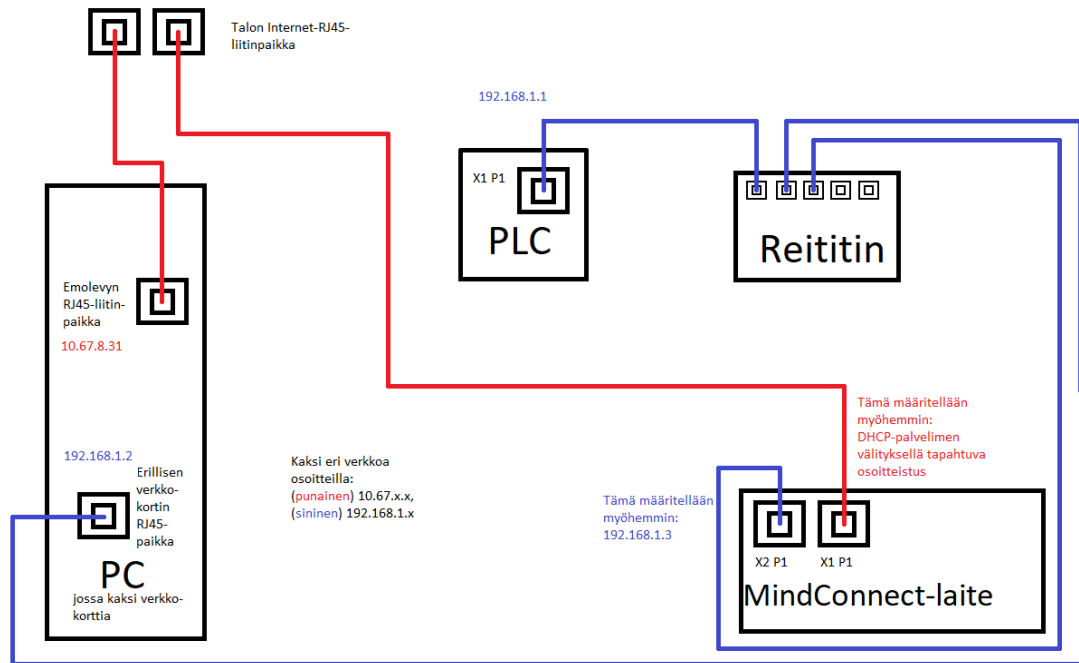
MindConnect IoT2040 -laitetta, joka näkyy kuvassa 6, käytetään datan keräykseen ja sen siirtoon MindSphereen ja sopii pieniin projekteihin. Laite lähettää Nanon tavoin salattua dataa turvatus Internet-yhteyden välityksellä MindSphereen.[6, s.19.] Laite pystyy käsittelemään 30 S7-protokollan data pointia sekunnissa [6, s.122]. Laitteen päällä sijaitsee 24 VDC:n jännitteensyöttöliitin sekä toimintatilaa näyttävät LED-valot. LED-valojen toiminnan ohjeet löytyvät Getting Connected to MindSphere -ohjeesta sivuilta 27 ja 28. Laitteen pohjasta, joka näkyy kuvassa 7, löytyy X60 USB-portti konfigurointitiedostojen latausta varten sekä RJ45-liitinpaikat verkkoyhteyksiä varten.



Kuva 7. MindConnect IoT2040 -laitteen pohja [6, s.43].

2.4 Internet- ja lähiverkkojen asennukset.

MindConnect-laitteet tarvitsevat kaksi eri verkkoa toimiakseen. Molemmissa laitteissa on PN/IE-liitinpaikka eli toisin sanoen kaksi RJ45-liitinpaikkaa, jotka on merkitty paikoiksi X1 P1 ja X2 P1. X1 P1 -liittimen verkko on suoraan yhteydessä Internetiin joko suoraan seinästä tulevan yhteyden välityksellä tai reitittimen tai kytkimen kautta. X2 P1 -liittimen verkko on se lähiverkko, jossa MindSphereen syötettävien muuttujien PLC-logiikka sijaitsee. X1 P1 -liittimen verkon ja X2 P1 -liittimen verkon on oltava eri verkkoja. Esimerkiksi jos X1 P1 -liittimen verkon osoite on 192.168.1.1, niin X2 P1 -liittimen verkon osoite pitää olla 192.168.2.1.[6, s.46.] Kuvassa 8 nähdään esimerkkikytkennät.

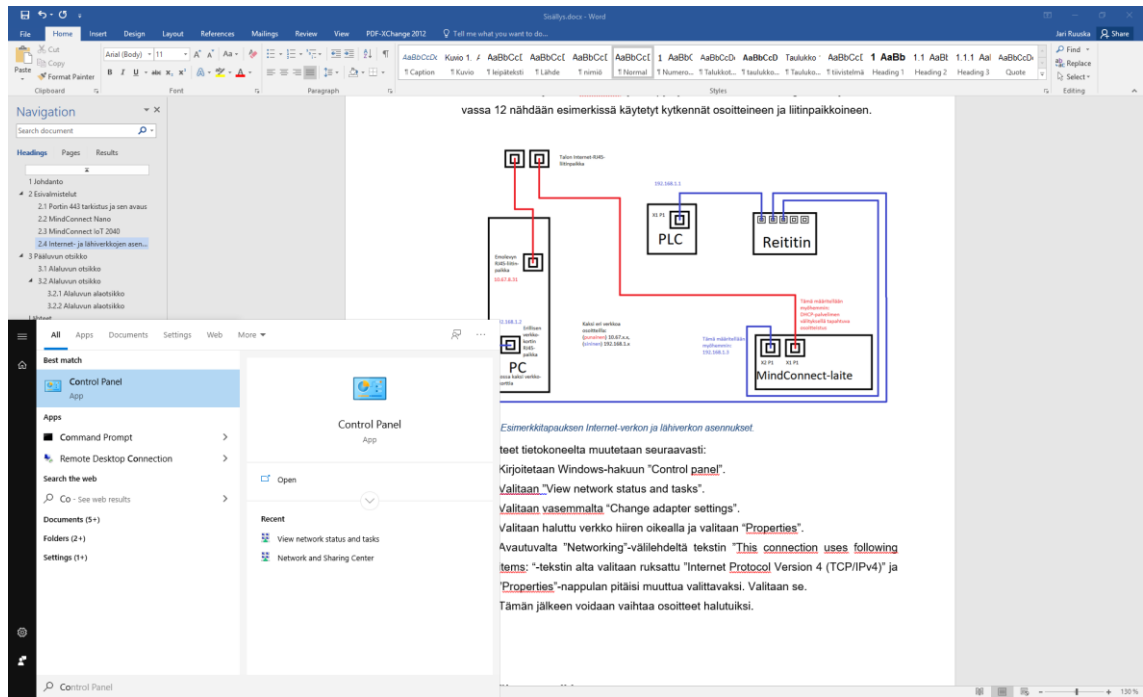


Kuva 8. Esimerkitapauksen Internet-verkon ja lähiverkon asennukset.

2.4.1 IP-osoitteiden vaihto tietokoneelta

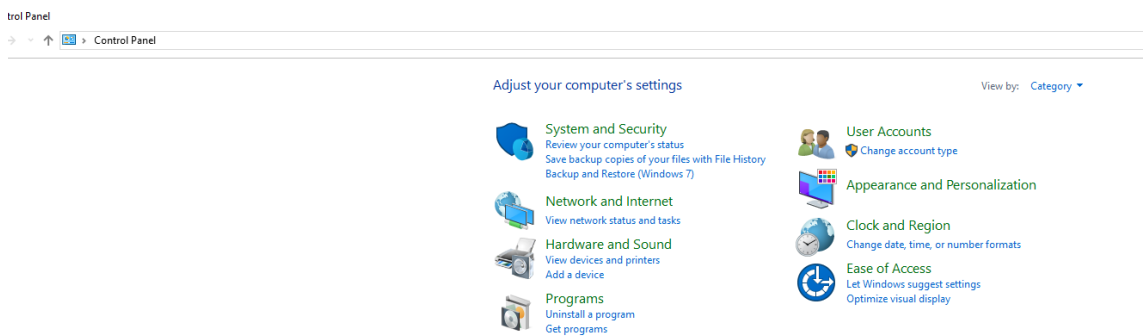
IP-osoitteet tietokoneelta muutetaan seuraavasti:

- Kirjoitetaan Windows-hakuun Control Panel ja valitaan se (kuva 9).



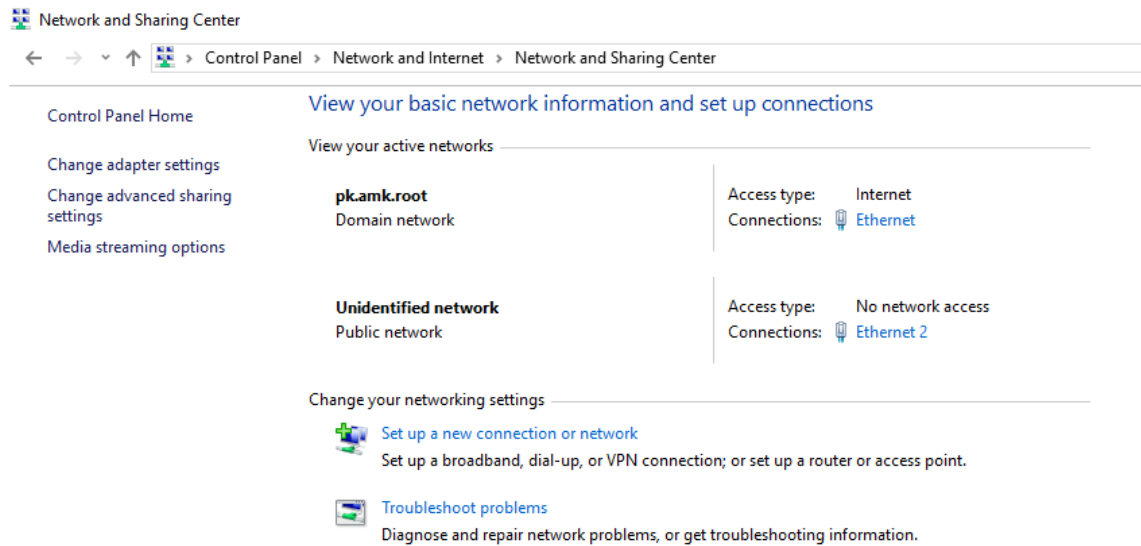
Kuva 9. Ohjauspaneelin etsintä.

- Valitaan View network status and tasks (kuva 10).



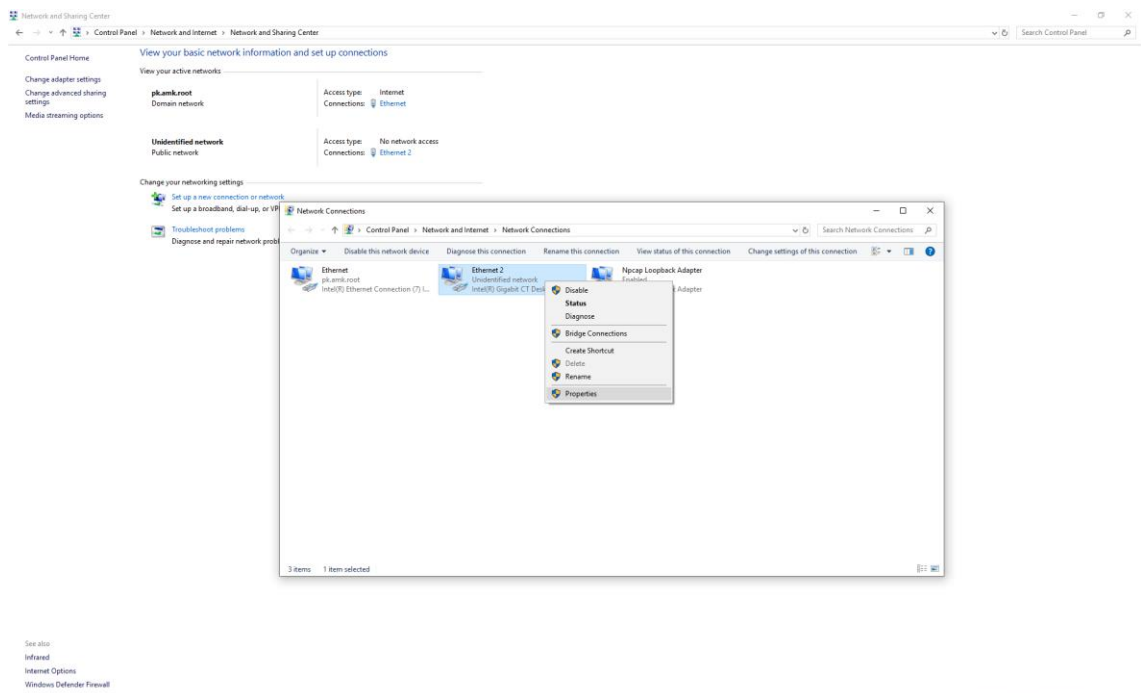
Kuva 10. Ohjauspaneeli.

- Valitaan vasemmalta Change adapter settings (kuva 11).



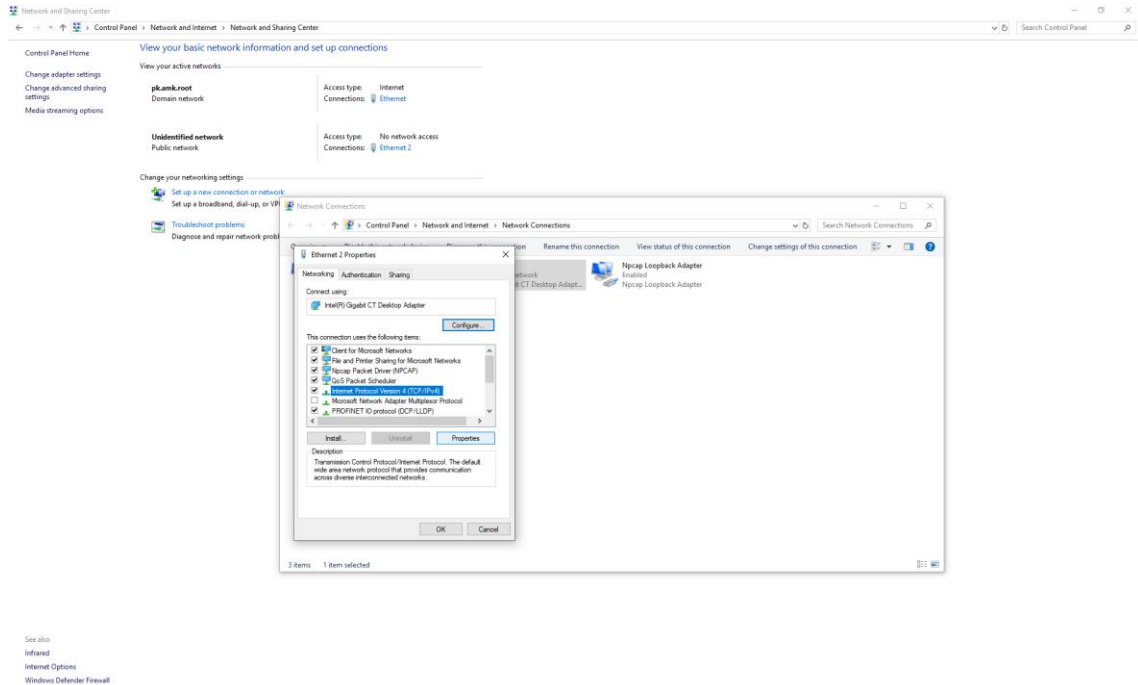
Kuva 11. Verkko- ja jakamiskeskus.

- Valitaan haluttu verkko hiiren oikealla ja valitaan Properties (kuva 12).



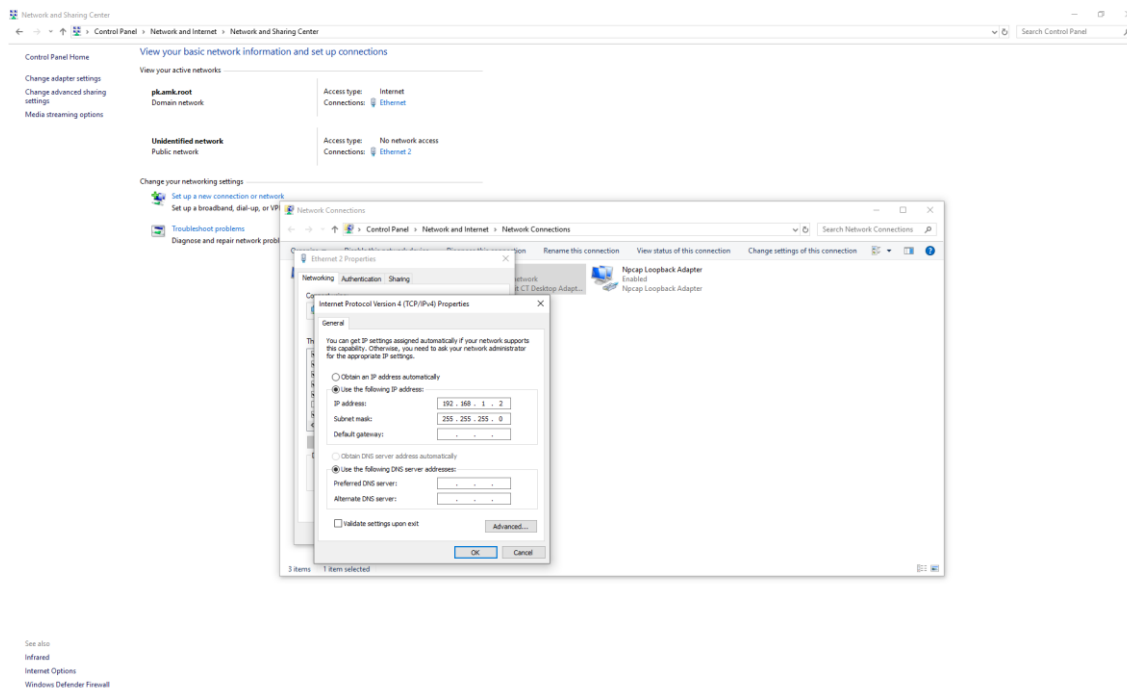
Kuva 12. Halutun verkon valinta. Esimerkkitapauksessa valittiin kuvan 8 mukaisen sinisen verkon verkkokortti.

- Avautuvalta Networking-välilehdellä nähdään keskellä yhteystapoja, joista valitaan ruksattu Internet Protocol Version 4 (TCP/IPv4) ja Properties-nappulan pitäisi muuttua aktiiviseksi. Valitaan se (kuva 13).



Kuva 13. Verkon yhteysmahdollisuudet.

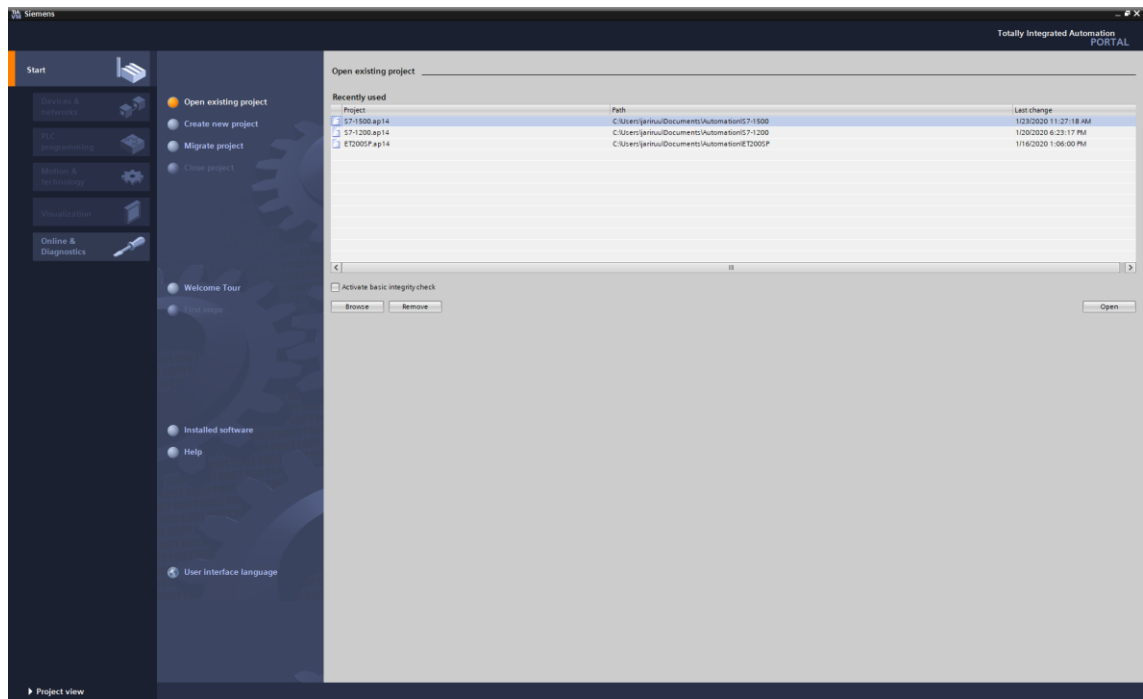
- Tämän jälkeen voidaan vaihtaa osoitteet halutuiksi (kuva 14). Muutoksien jälkeen valitaan OK.
 - o IP-address muuttaa verkon IP-osoitteen
 - o Subnet mask muuttaa aliverkon maskin.
 - o Default gateway muuttaa oletusyhdyskäytävän



Kuva 14. IP-asetuksien muuttaminen.

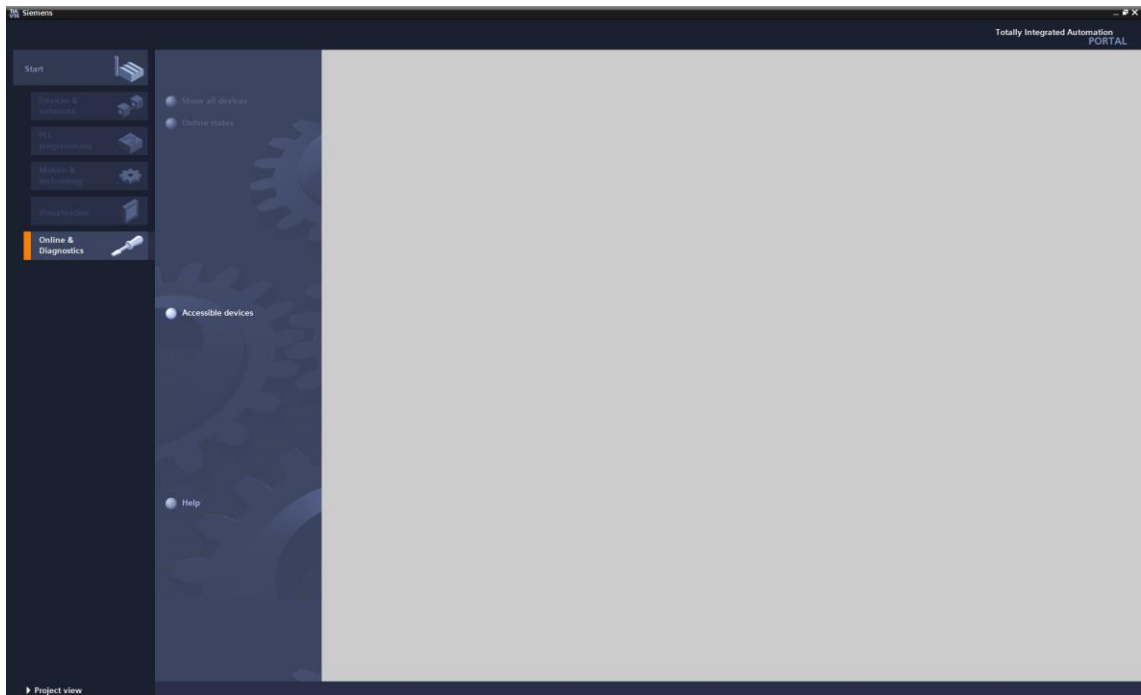
2.4.2 IP-osoitteiden vaihto PLC-logiikalle

- TIA Portalin alkuruudusta valitaan vasemmalta valikosta Online & Diagnostics (kuva 15).



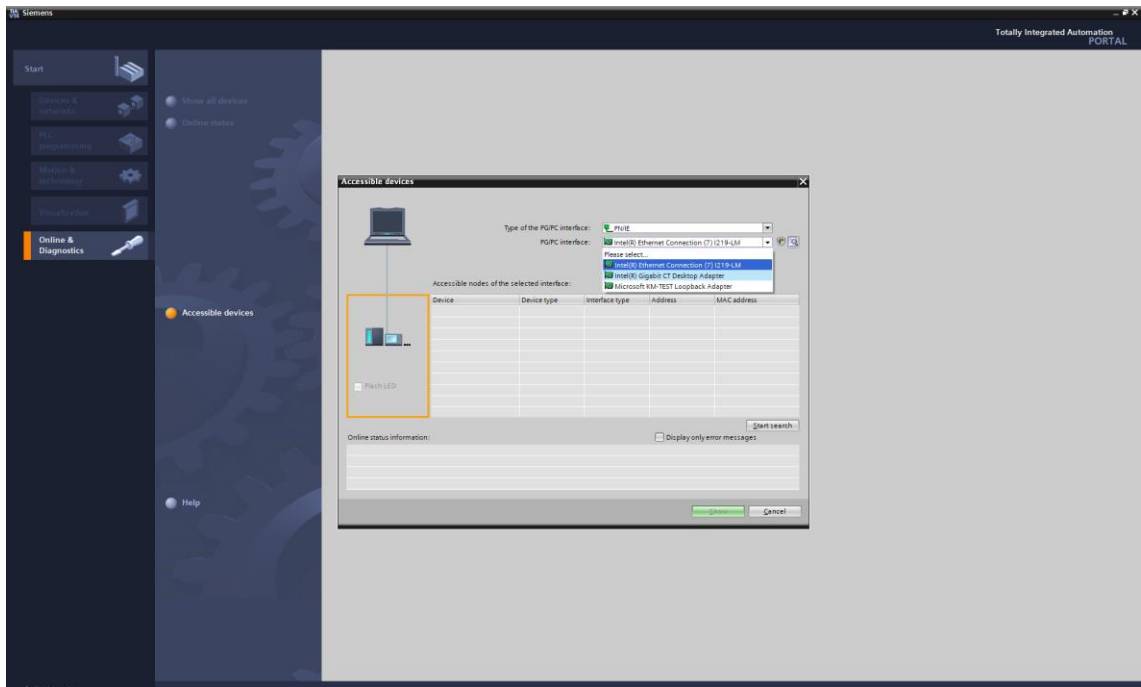
Kuva 15. TIA Portalin alkuruutu.

- Valitaan Accessible devices (kuva 16).



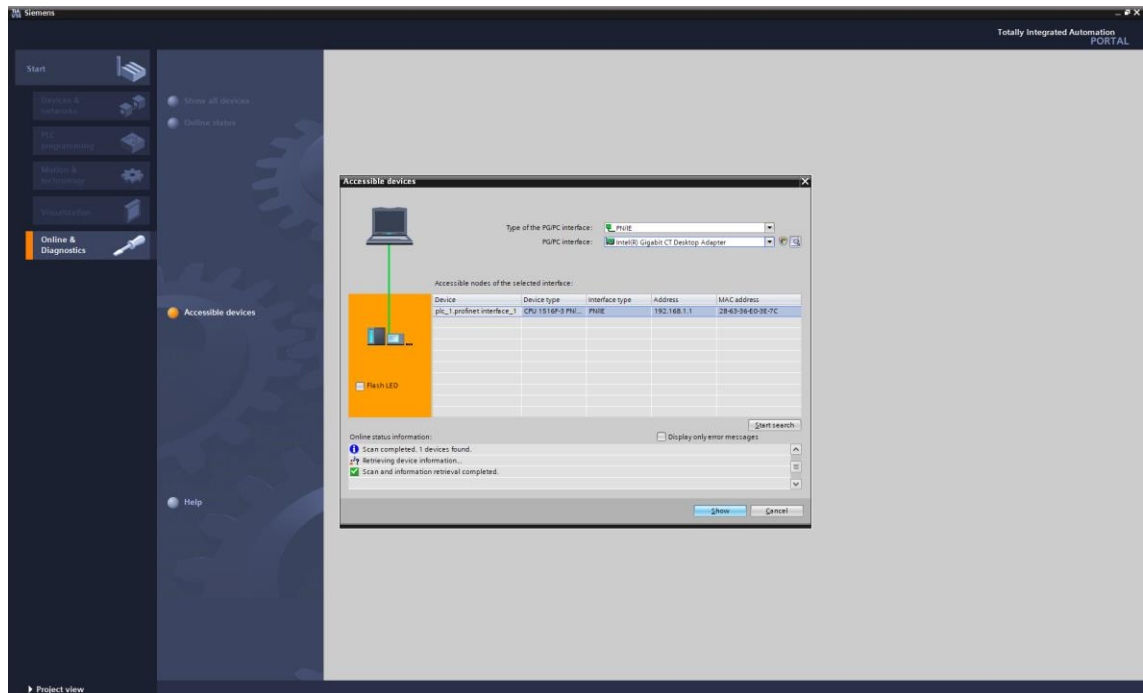
Kuva 16. Online & Diagnostics -ikkuna.

- Valitaan PG/PC Interface -kohtaan se yhteystapa, jolla tietokone on yhteydessä PLC-logiikkaan. Valitun tavan jälkeen painetaan Start search (kuva 17).



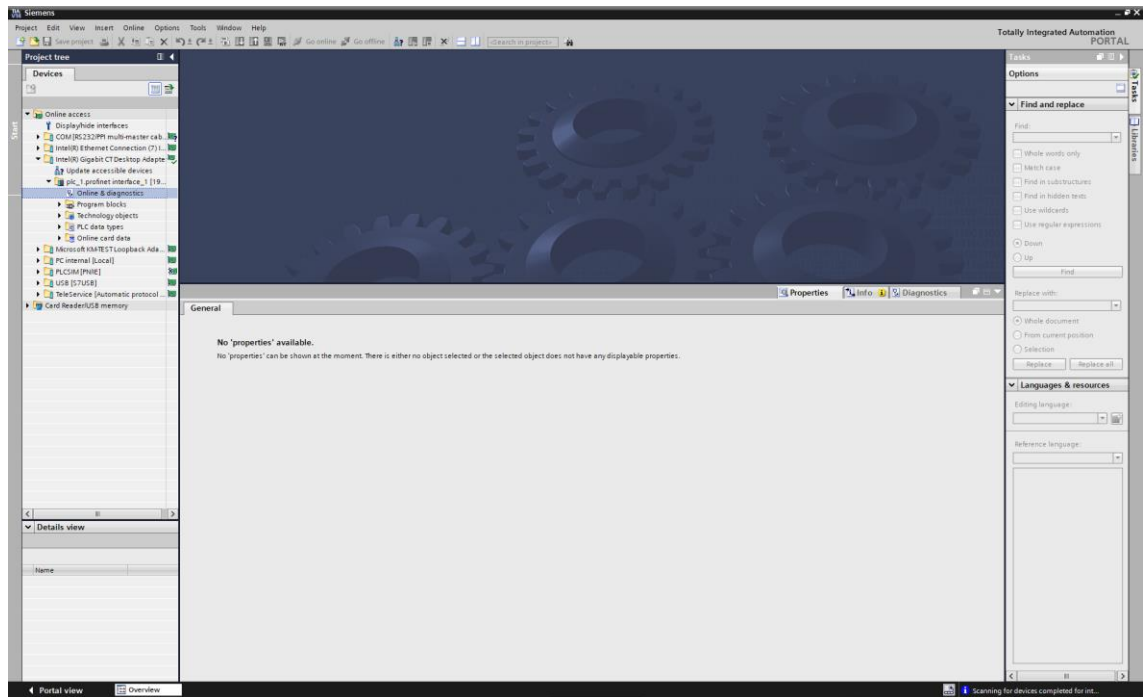
Kuva 17. Accessible devices -hakutoiminto.

- Haun pitäisi etsiä verkkoon kytketyt laitteet, esimerkkitapauksessa kuvan 18 mukaan PLC-logiikka löytyy osoitteesta 192.168.1.1. Valitaan Show-painike.



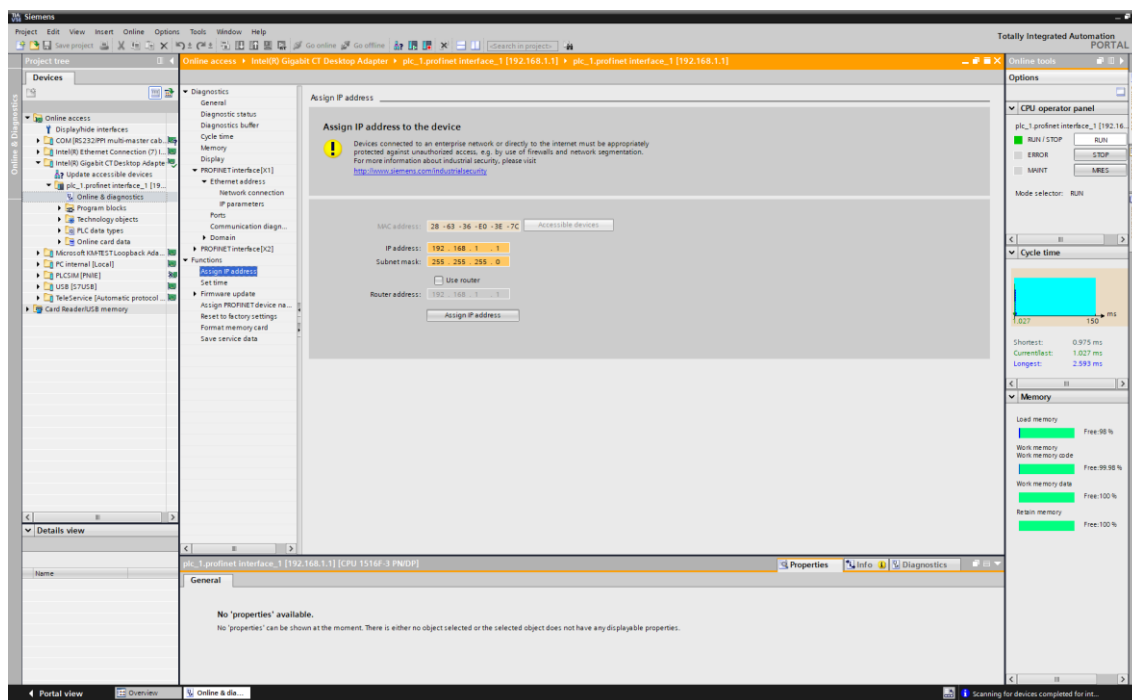
Kuva 18. Accessible devices -hakutoiminto, joka on löytänyt verkossa olevan PLC-logiikan.

- Valitaan avautuvasta projektinäkömästä vasemmalta puuvalikosta avatun PLC-logiikan Online & diagnostics -välilehti (kuva 19).



Kuva 19. TIA Portalin projektinäkymä.

- HUOM! Tätä ei kannata tehdä, mikäli ei tiedä mitä on tekemässä. Aluksi pysytetään online-tilassa oleva PLC-logiikka ruudun oikealla olevasta CPU operator panel -kohdasta painamalla STOP-näppäintä. Tämän jälkeen valitaan välilehdeltä keskikohdan vasemmalta puolelta puuvalikosta Functions, jonka alta valitaan Assign IP address. Täältä välilehdeltä voidaan vaihtaa aikaisemmin etsityn PLC-logiikan IP-osoite ja aliverkon maski. Lopuksi käyttäjän pitää varmistukseen vaihdot painettava välilehdellä olevaa Assign IP address -painiketta (kuva 20).



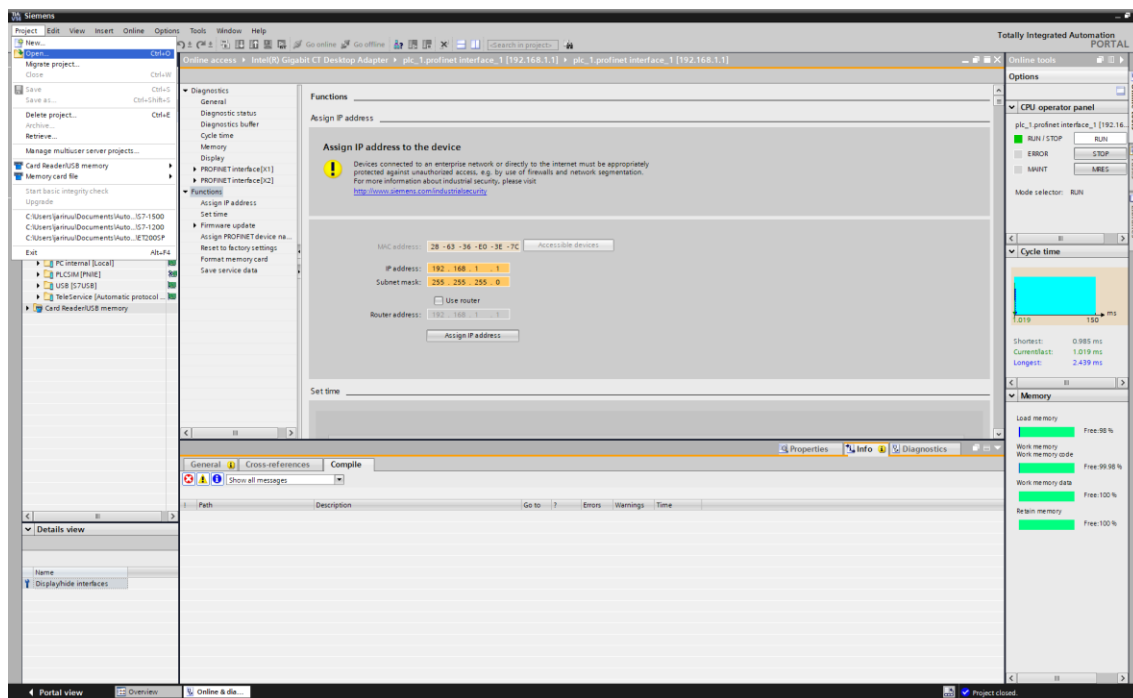
Kuva 20. Assign IP address -välilehti.

- TIA Portal hukkaa logiikan, jonka jälkeen logiikka on etsittävä uudestaan. Mikäli vaihdettu osoite on samassa verkossa kuin edellinen osoite eli esimerkiksi 192.168.1.1 vaihtuu 192.168.1.2-osoitteeksi, niin valitaan aluksi TIA Portal -näkyvän vasemmasta alareunasta Portal view. Tämän jälkeen avautuvasta Online & Diagnostics -näkymästä valitaan Accessible devices -painike. Haun pitäisi löytää logiikka, kuten aiemminkin.
- Mikäli osoite vaihtuu joksikin muuksi esimerkiksi 192.168.1.1 vaihtuu 192.168.2.1-osoitteeksi niin silloin pitää aluksi käydä muuttamassa koneen verkkokortin osoite (ks. IP-osoitteiden vaihto tietokoneelta) samaan verkkoon kuin vaihdettu PLC-logiikan osoite. Esimerkiksi osoitteeksi 192.168.2.2. Tämän jälkeen voidaan etsiä TIA Portalista Accessible devices -hakutoiminnolla PLC-logiikkaa.
- Mikäli PLC-logiikka näyttää IP-osoitteiden vaihdon jälkeen MAINT-vikaleidiä, niin tähän auttaa PLC-logiikan uudelleenkäynnistys.

- Lisäksi käyttäjän on huomioitava TIA Portalista oman projektinsa PLC-logiikan osoite samaksi kuin muutettu fyysisen PLC-logiikan osoite. Muuten käyttäjä ei voi ladata omaa ohjelmaansa PLC-logiikalle. Tästä lisää kohdassa Oman TIA-projektin IP-osoitteiden vaihto.

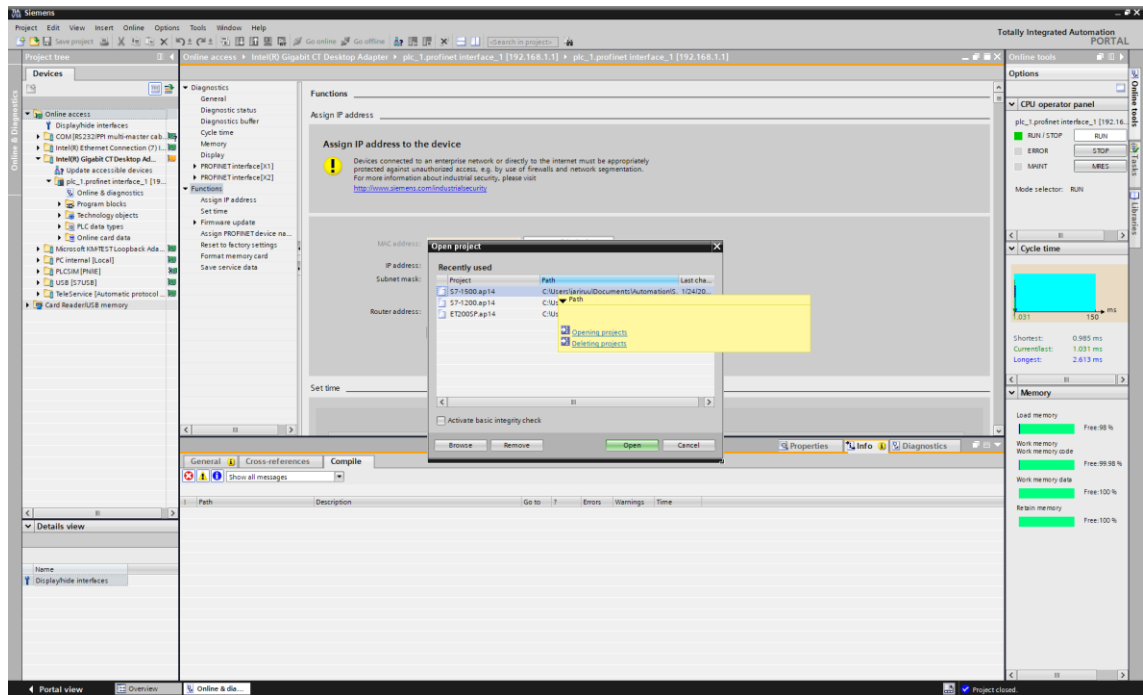
2.4.3 Oman TIA-projektin IP-osoitteiden vaihto

- Valitaan vasemmalta ylävalikkoriviltä Project ja avautuvasta pudotusvalikosta Open (kuva 21).



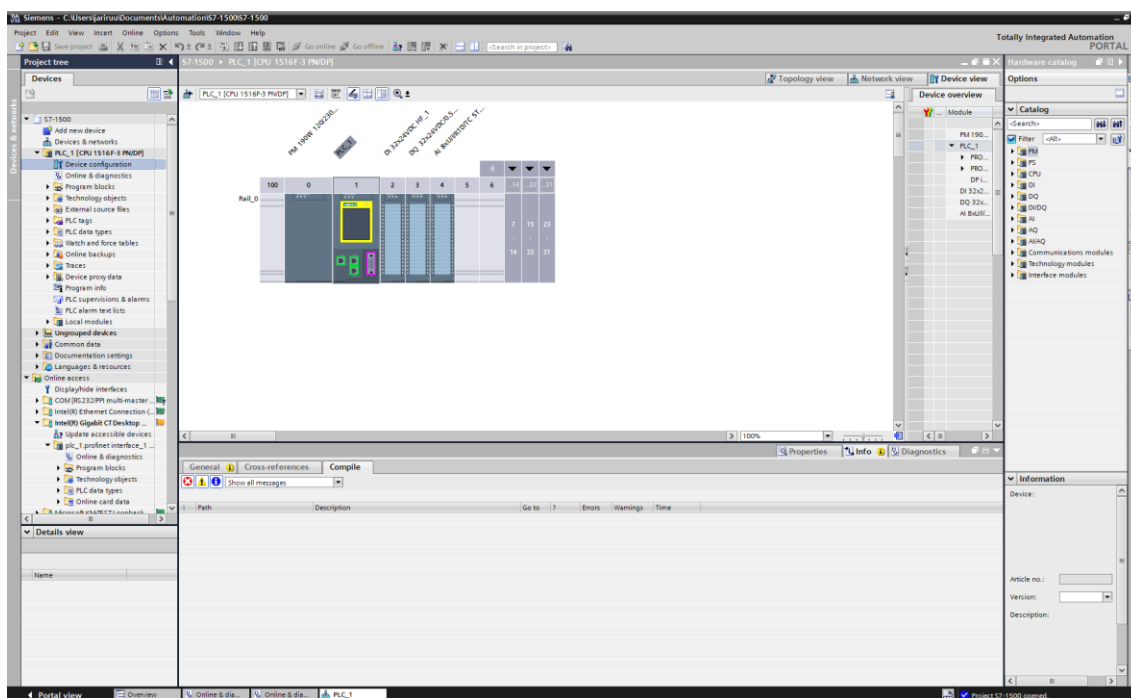
Kuva 21. Oman projektin avaus TIA Portalista.

- Valitaan avautuvasta ikkunasta haluttu projekti ja painetaan Open (kuva 22).



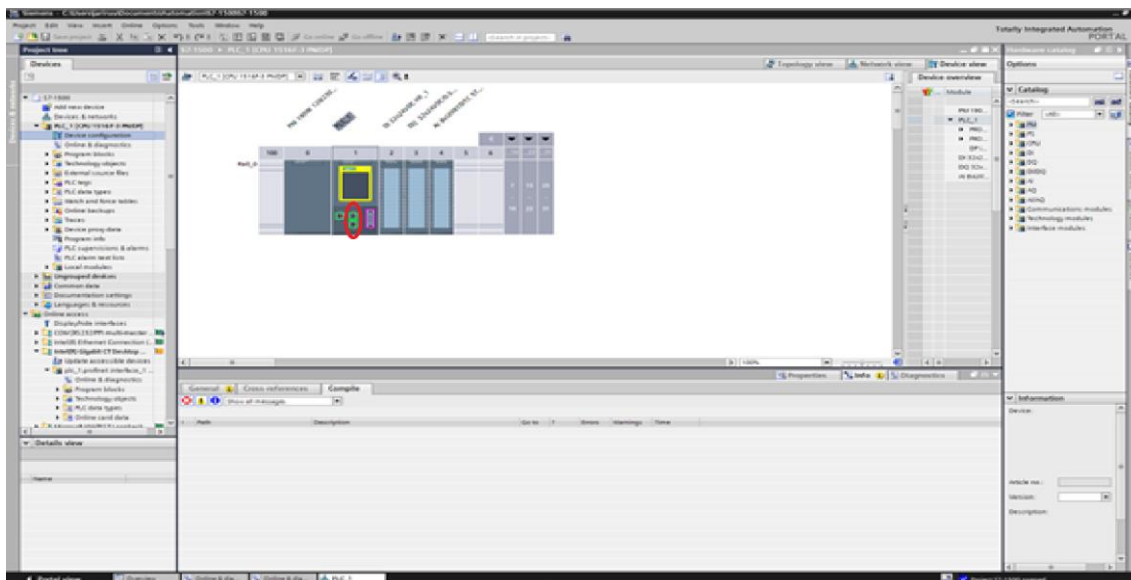
Kuva 22. Open project -valikkoruutu.

- Valitaan vasemmalta puuvalikosta oman projektikansion alta Device configuration. Kuvassa 23 valinta on maalattu sinisellä.



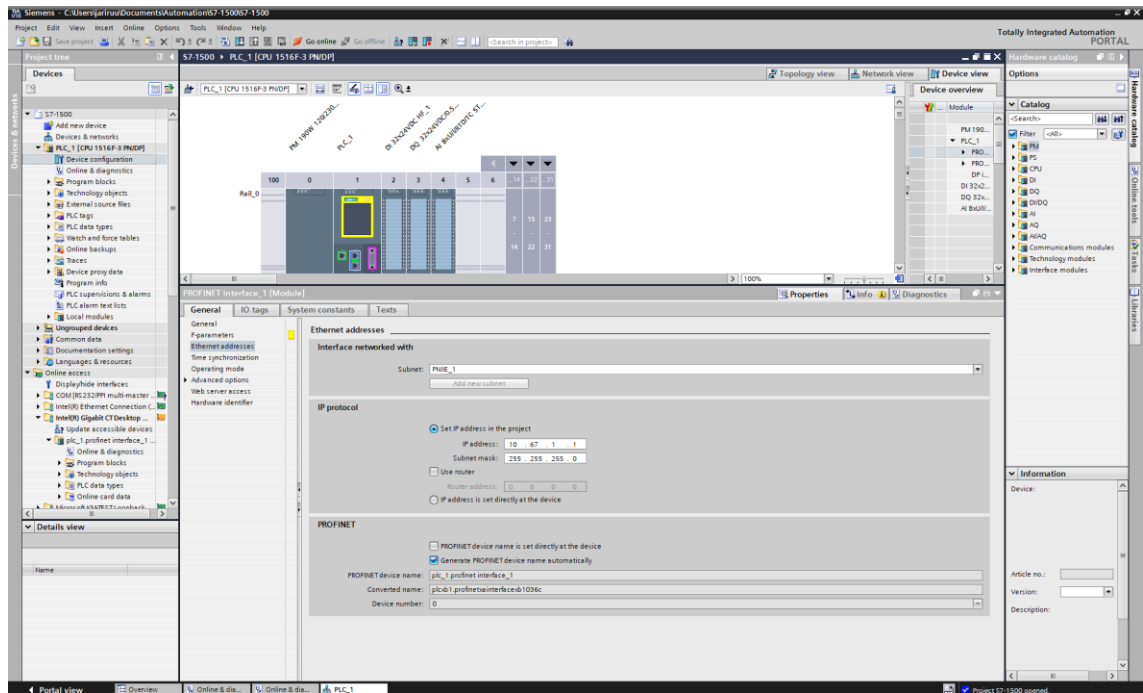
Kuva 23. Oman projektin Device configuration -valinta.

- Valitaan oikea PN/IE-portti, esimerkkitapauksessa PLC-logiikan X1-portti, joka on merkitty punaisella ellipsillä kuvassa 24.



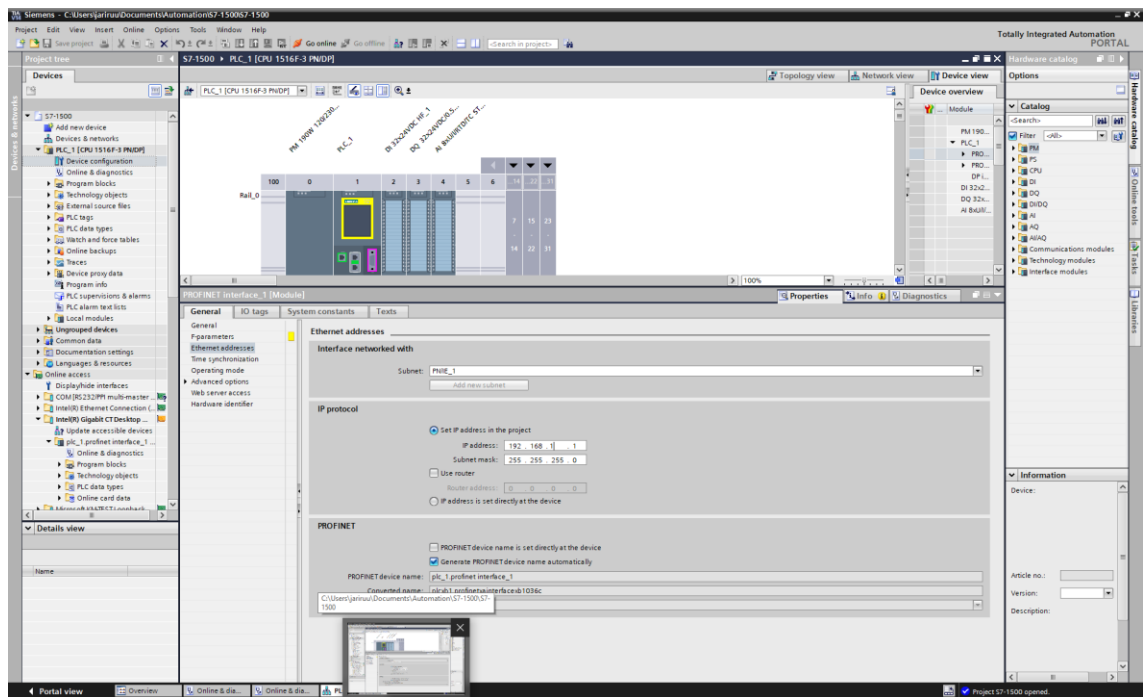
Kuva 24. Oikean PN/IE-portin valinta PLC-logiikasta.

- Valitaan avautuvasta valikosta Ethernet addresses (kuva 25).



Kuva 25. IP-osoitteen vaihto PLC-projektista.

- Mikäli projektin IP-osoite on eri kuin fyysisen PLC-logiikan osoite, muutetaan se samaksi. Vaihto tapahtuu välilehden kohdasta Set IP address in the project -kohdasta mihin kirjoitetaan uusi osoite. Tämän jälkeen kannattaa tallentaa projekti. Esimerkitapauksessa 10.67.1.1 muutetaan 192.168.1.1, joka on fyysisen PLC-logiikan osoite (kuva 26).
- Tämän jälkeen projektin lataaminen PLC-logiikalle pitäisi onnistua.

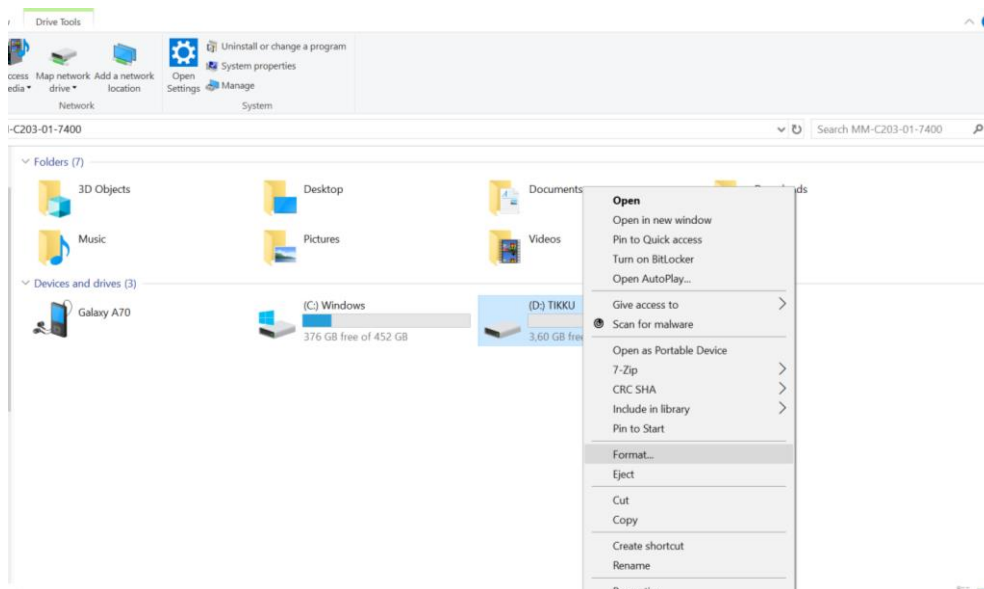


Kuva 26. IP-osoite vaihdettu fyysisen PLC-logiikan mukaiseksi.

2.5 USB-tikun alustaminen FAT32-tiedostomuotoon

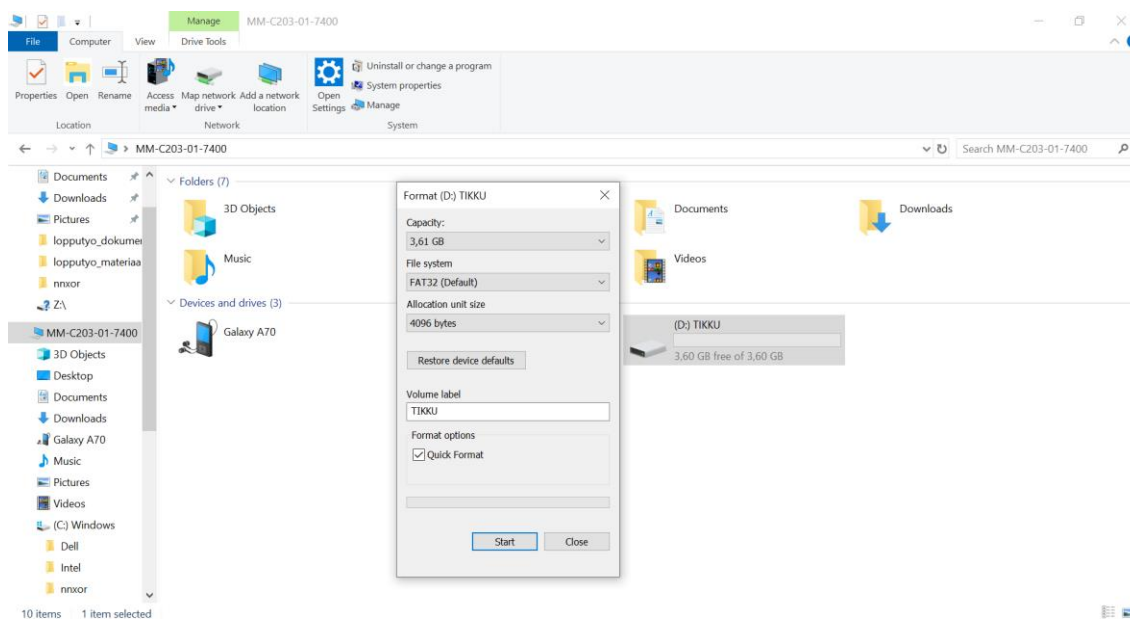
MindConnect-laitteen käyttöönottoon tarvitaan USB-tikku, joka on alustettu FAT32-tiedostomuotoon. Tikun pitää olla tyhjä, jotta MindSpheren käyttöönotto onnistuu. USB-tikku on oletuksena FAT32-tiedostomuodossa, joten tämä vaihe on vain tarvittaessa.

- USB-tikun alustus tapahtuu siten, että USB-tikun ollessa kiinni tietokoneen USB-portissa avataan Omasta tietokoneesta USB-tikun kansio hiiren oikealla ja valitaan kohta Format (kuva 27).



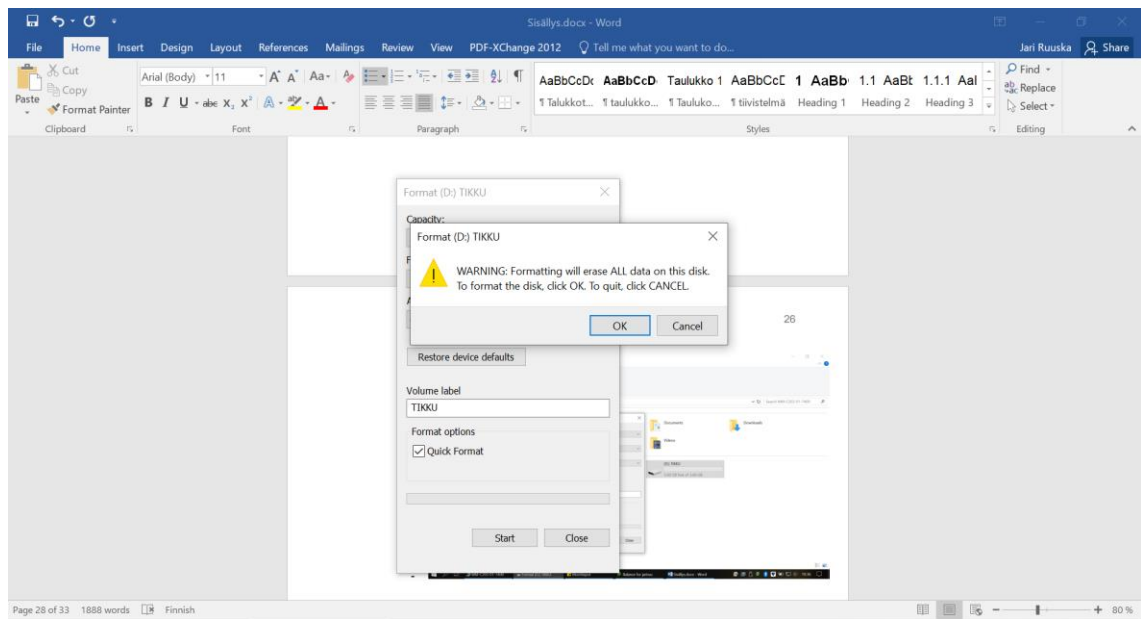
Kuva 27. USB-tikun alustus Omasta tietokone-ikkunasta.

- Tämä avaa valintaruudun, josta valitaan kuvassa 28 näkyvät valinnat.



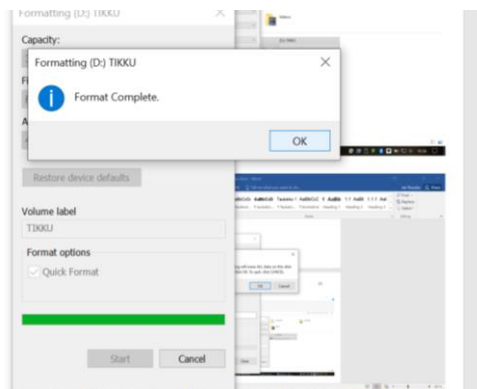
Kuva 28. Alustus-ohjelma.

- Ohjelma kysyy, että tehdäänkö alustus. Tähän OK (kuva 29).



Kuva 29. Alustuksen varmistuskysymys.

- Ohjelma näyttää lopuksi ruudun, missä kerrotaan alustuksen onnistuneen. Tämä poisti tikulla olleet tiedostot ja asetti tikun haluttuun FAT32-muotoon (kuva 30).

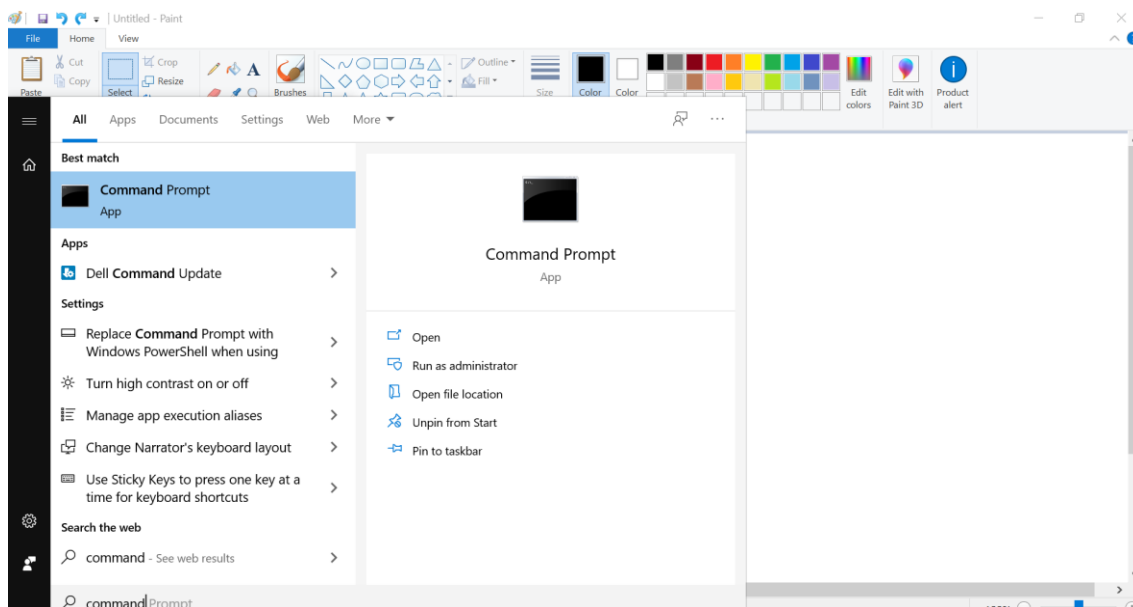


Kuva 30. Alustuksen onnistumisteksti.

2.6 Portin 443 tarkistus ja sen avaus

Portti 443 on lähtökohtaisesti auki ulkopuolelta tulevan Internet-liikenteen takia. Tässä osuudessa kerrotaan, että miten portin tila voidaan havaita ja miten sen avaus tapahtuu.

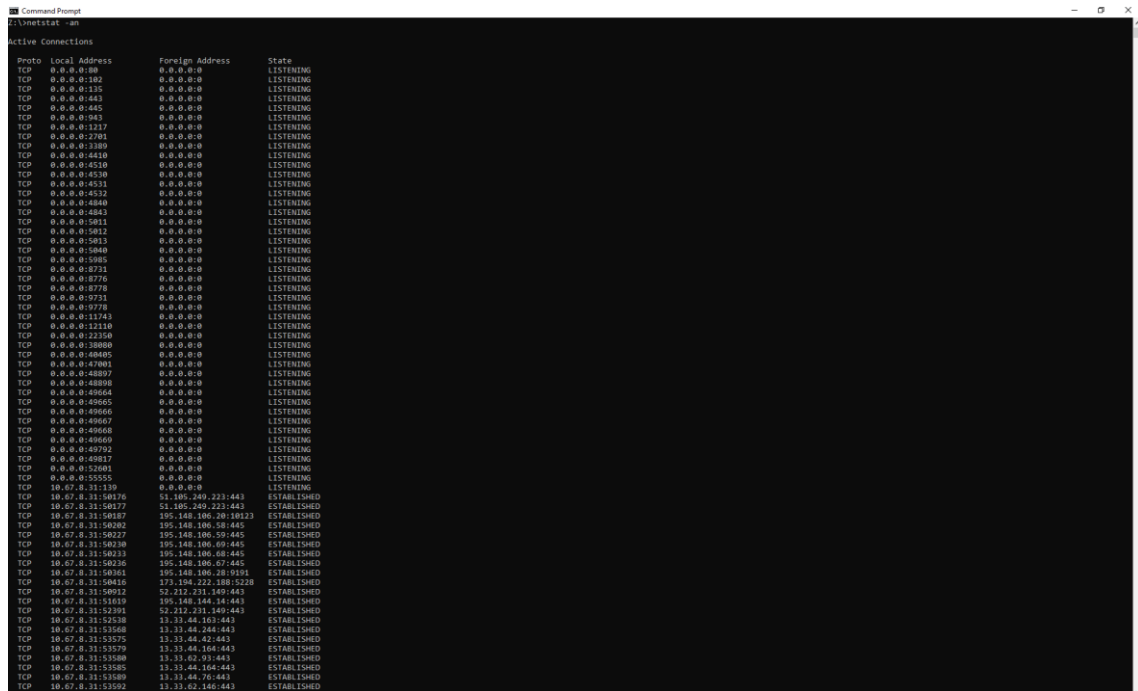
- Avataan Command Prompt -ohjelma Windows-valikosta (kuva 31).



Kuva 31. Komentokehoitteen haku Windows-valikosta.

- Kirjoitetaan komentoriville **netstat -an**.

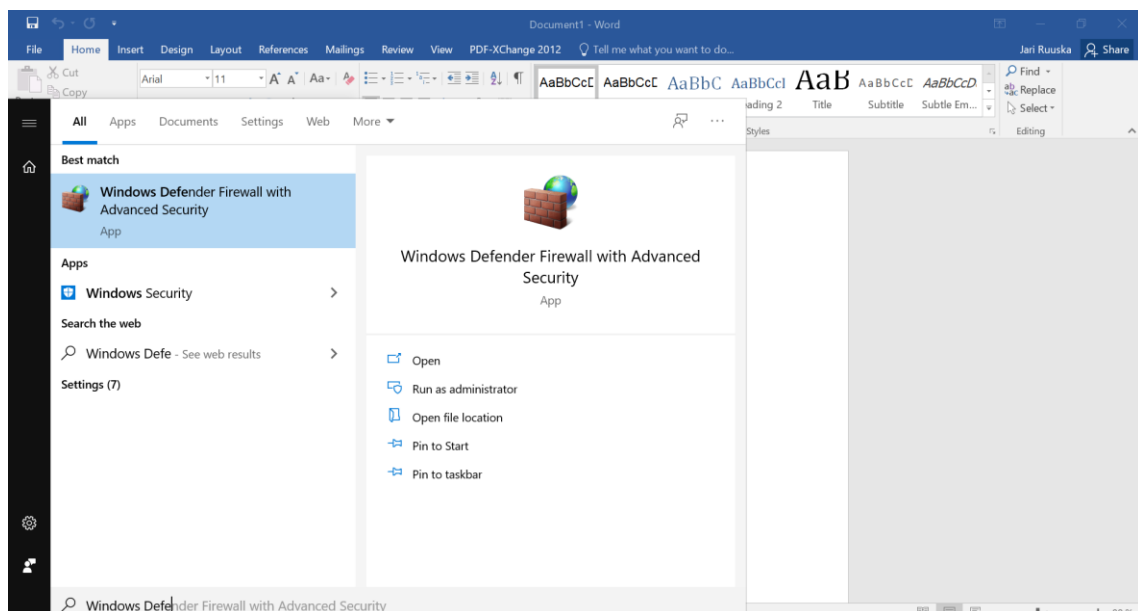
Tämä komento antaa käyttäjälle tiedon siitä, että mitkä portit ja osoitteet TCP- ja UDP-yhteyksistä ovat toiminnassa ja mitkä porteista ovat auki. Väliviivan jälkeinen a-kirjain näyttää kaikki aktiiviset portit ja n-kirjain näyttää osoitteet ja porttinumerot numeerisesti. [7.] Komento antaa rivejä, joissa on neljä saraketta. Ensimmäinen sarake kertoo mikä protokolla on kyseessä, toinen sarake kertoo paikallisen lähiverkon osoitteen ja sen portin kaksoispisteiden oikealla puolella, kolmas sarake kertoo lähiverkon ulkopuolisen osoitteen eli millä osoitteella lähiverkon osoite lähettää paketteja Internetiin ja sen portin kaksoispisteiden oikealla puolella ja neljäs sarake kertoo osoitteen tilan eli onko osoite toiminnassa. Portti 443 näkyy netstat-toiminnon Foreign Address -sarakkeessa, kuten esimerkiksi kuvassa 32, jossa ensimmäinen TCP-protokollan osoite, joka on muuta kuin 0.0.0.0 näyttää osoitteeksi 51.105.249.223:443 eli protokollan IP-osoitteen lähiverkon ulkopuolella ja sen portin. Tämä näyttää sarakessa State olevan toiminnassa.



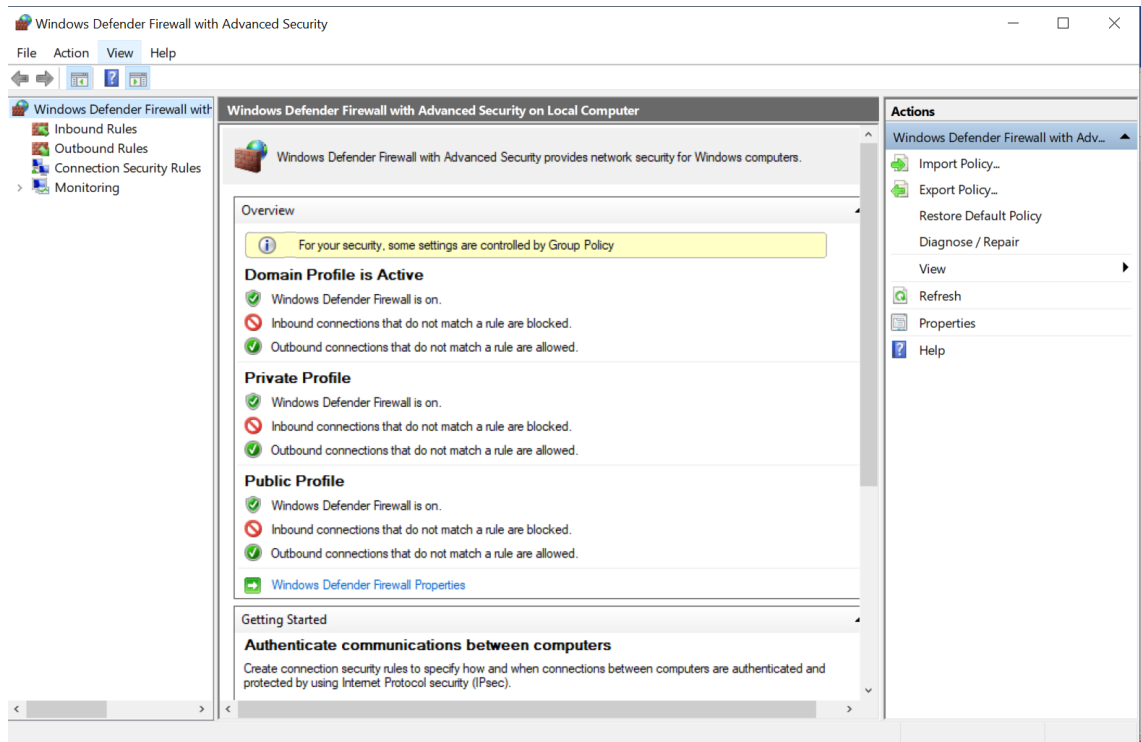
Kuva 32. Kuvan alalaidassa näkyy muutama osoiterivi, joissa portti 443 on toiminnassa.

Mikäli portti olisi jostain syystä kiinni voidaan se avata seuraavasti:

- Avataan Windows-valikosta Windows Defender Firewall with Advanced Security -ohjelma (kuva 33).
- Valitaan vasemmalta puukaaviosta Outbound Rules -väilihehti (kuva 34).

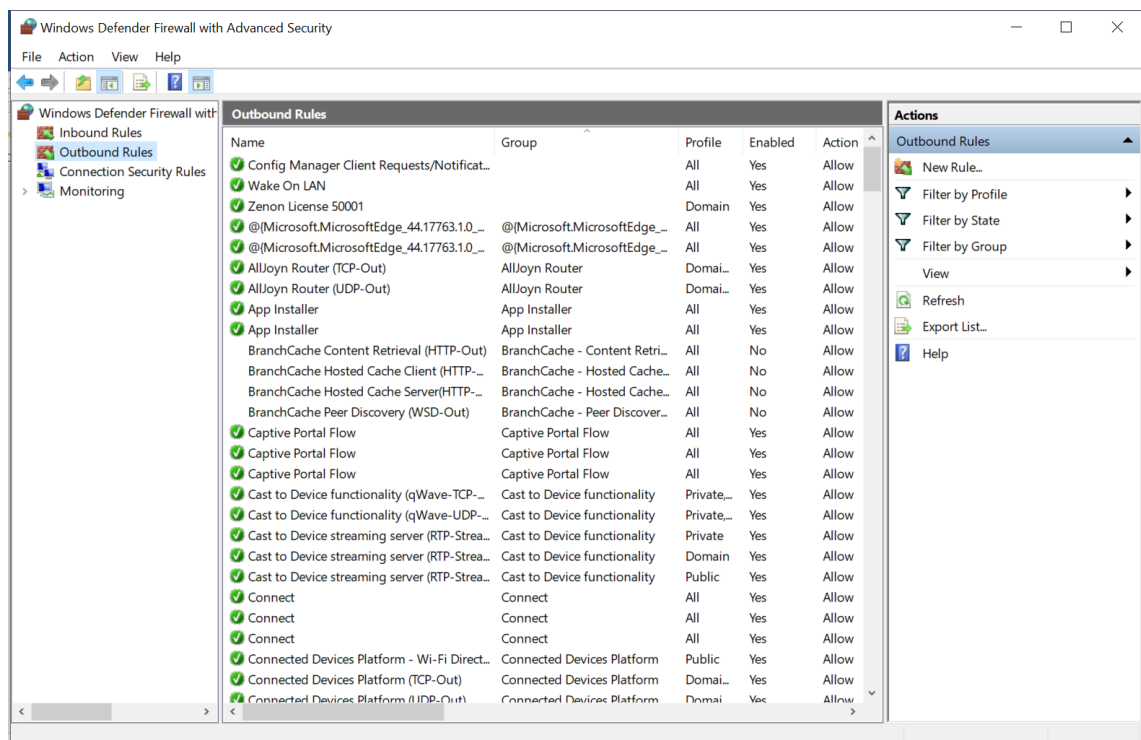


Kuva 33. Palomuurin etsintä Windows-valikosta.



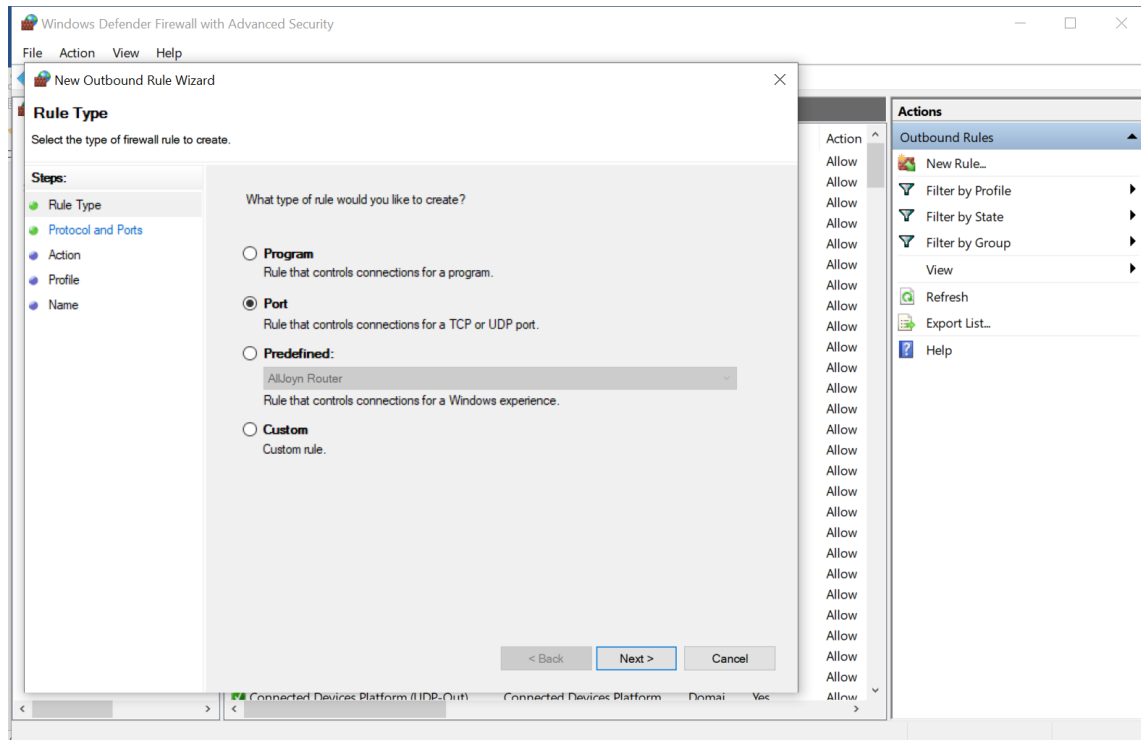
Kuva 34- Windows Defender Firewall with Advanced Settings-ohjelman päänäyttö.

- Välilehden oikealla olevasta Actions-valikosta valitaan New Rule (kuva 35).



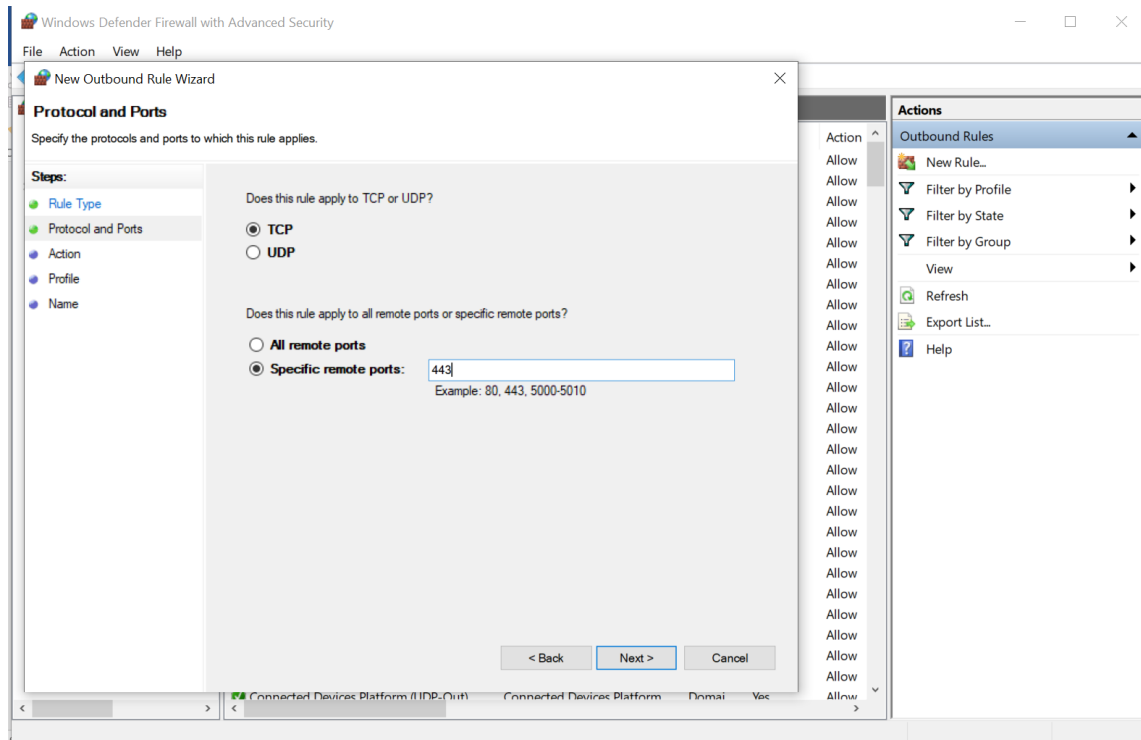
Kuva 35. Outbound Rules -välilehti.

- Avautuvasta velhotyökalusta valitaan vaihtoehto Port ja painetaan Next (kuva 36).



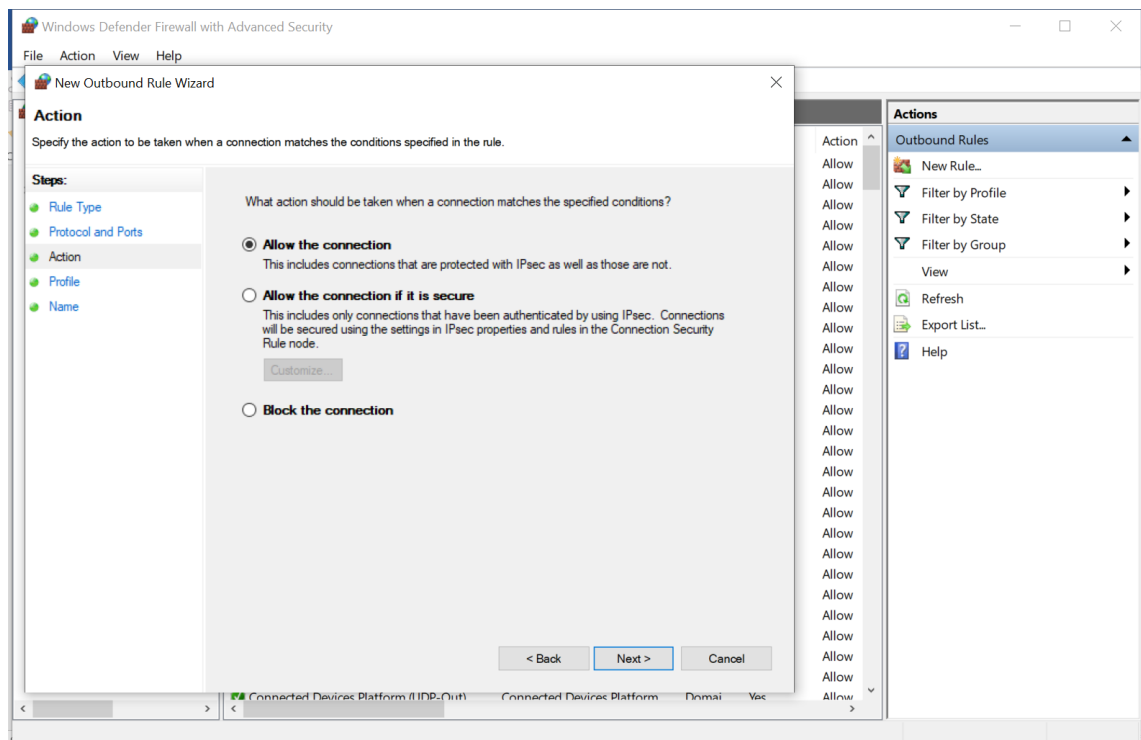
Kuva 36. Velhotyökalun säännön tyypin valinta.

- Seuraavassa kohdassa annetaan protokollan olevan TCP, mutta valitaan säännön koskevan jotain tiettyä porttia eli valitaan Specific remote ports ja kirjoitetaan viereiseen kenttään 443. Mennään eteenpäin valitsemalla Next (kuva 37).



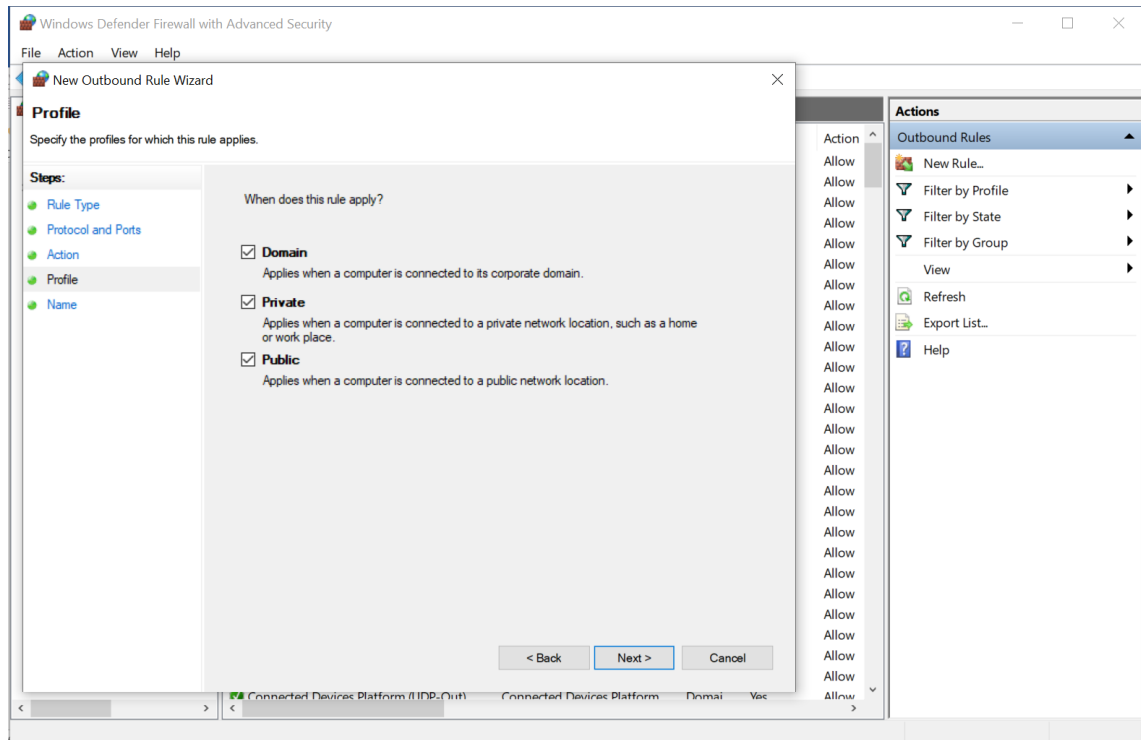
Kuva 37. Protokollan ja portin valinta.

- Sitten sallitaan yhteyden tapahtuvan eli valitaan vaihtoehto Allow the connection. Painetaan Next (kuva 38).



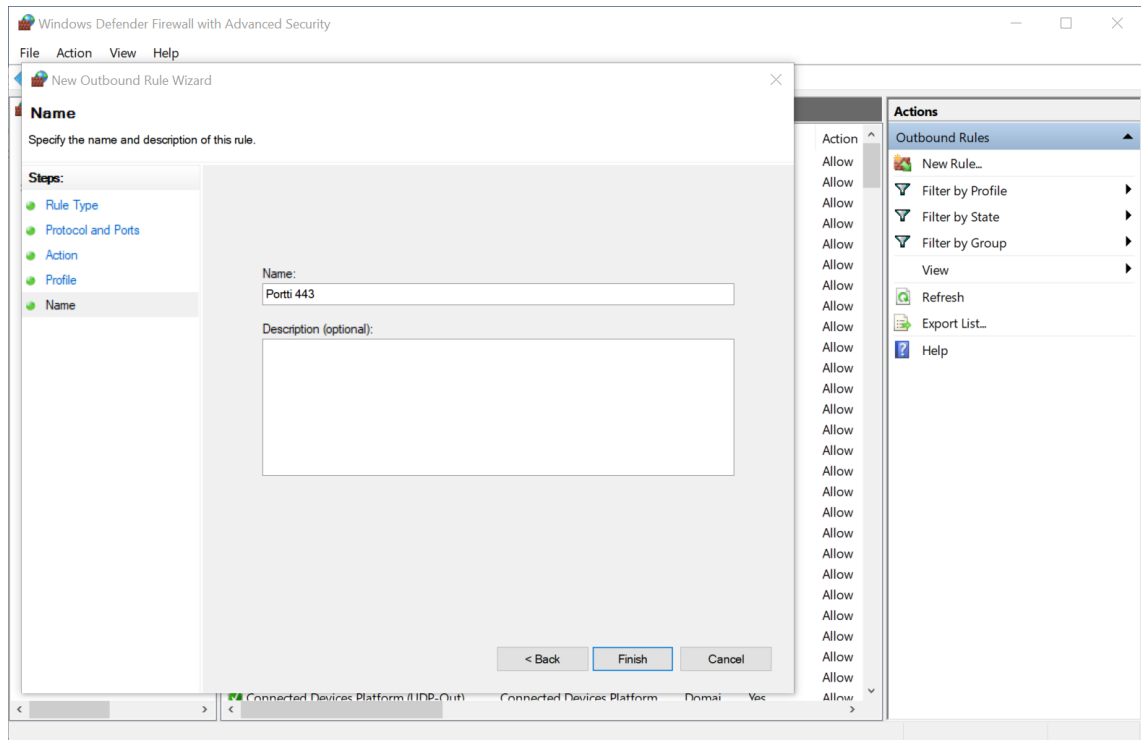
Kuva 38. Velhon kohdassa valitaan mitä portin yhteydelle tehdään.

- Tämän jälkeen voidaan valita, minkälaisia yhteyksiä sääntö koskee. Esimerkitapauksessa sääntö koskee kaikkia yhteyksiä. Valitaan Next (kuva 39).



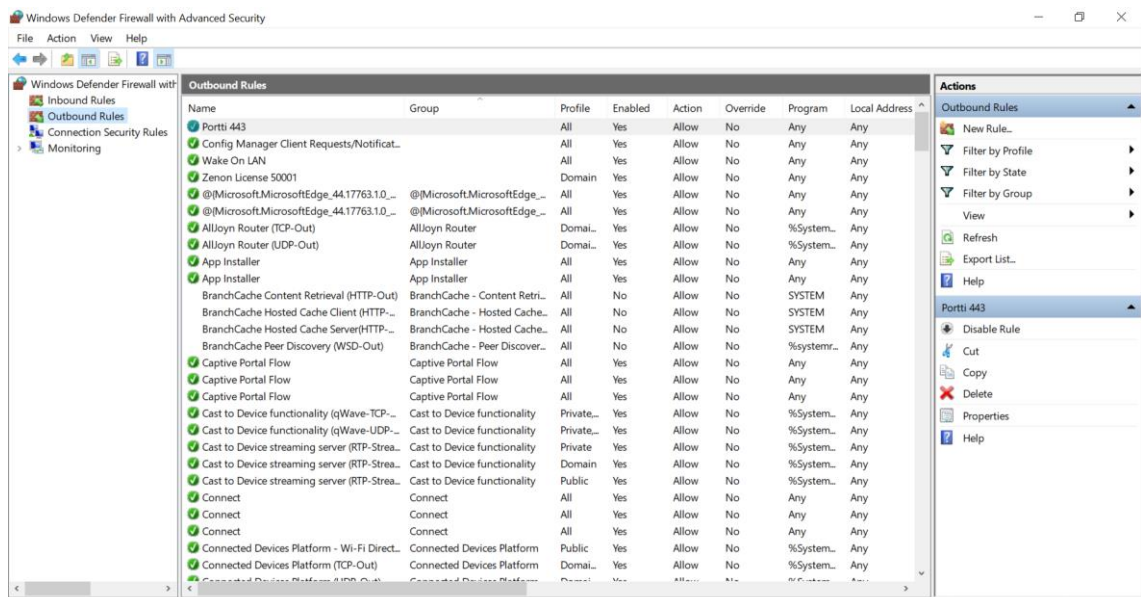
Kuva 39. Velhossa valitaan mitä yhteyksiä sääntö koskee.

- Lopuksi annetaan säännölle nimi ja painetaan Finish (kuva 40).



Kuva 40. Velhossa nimetään sääntö.

- Säännön voi nähdä nyt Outbound Rules -välilehden listan ensimmäisenä. Portin yhteyden voi testata esimerkiksi komentokehoitteen **netstat -an** -käskyllä tai yrittämällä hakea selaimella jotain Internet-osoitetta (kuva 41).



Kuva 41. Outbound Rules -välilehdessä ensimmäisenä listalla voidaan nähdä luotu sääntö.

3 MindSpheren käyttöönotto

Kun esivalmistelut on tehty, voidaan aloittaa MindSpheren käyttöönotto. Tässä vaiheessa oletetaan, että esivalmistelut on tehty sekä lisäksi PLC-logiikka on päällä sekä pitää sisällään ohjelman, MindConnect-laite on päällä ja tarvittavat verkkoyhteydet löytyvät laitteen, logiikan ja Internet-yhteyden välillä. Tässä luvussa käytetään esimerkkiä, jonka avulla ohje on tehty. Esimerkissä MindSphere on otettu käyttöön käyttäen MindConnect Nano -laitetta. Luvun kohdat on tehty siinä järjestyksessä missä onnistunut MindSpheren käyttöönotto esimerkin tapauksessa on tehty.

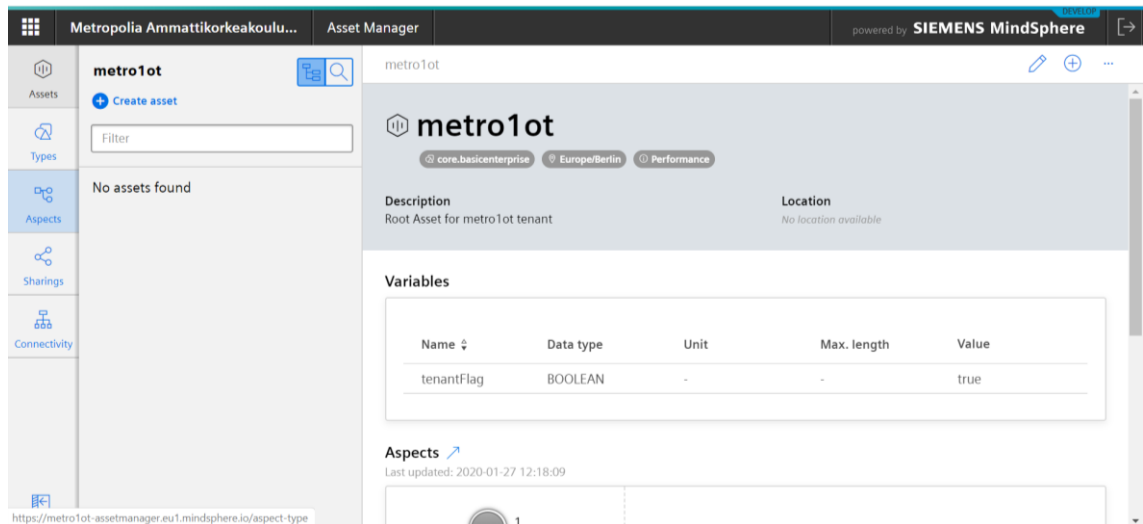
3.1 MindSphereen liittyvät käsitteet

Aspect	Aspect on nimitys MindSpheressä käytettävälle termille, joka toimii aspectin sisälle määriteltyjen muuttujien alustana, kuten esimerkiksi lamputila-aspect, jonka sisämuuttujat on nimetty nimillä sisa_lamputila ja ulko_lamputila. Aspectin sisämuuttujille ei ole määritelty määrää, vaan se voi sisältää useita muuttujia. [6, s. 60, 131.] Aspectin määrittämiseen kuuluu sen nimeäminen, sen sisältämien muuttujien luonti sekä aspectin asettaminen joko staattiseen tai dynaamiseen kategoriaan.
Assets	Assetit ovat fyysisten laitteiden tai automaatiojärjestelmien digitaalisia esityksiä, joiden automaatioyksikkö (tai -yksiköt), kuten esimerkiksi PLC-logiikka, on liitetty MindSphereen. Assetin sisältämä data kerätään ja lähetetään MindSphereen, jossa sen tarkastelua voidaan suorittaa esimerkiksi historiakuvaajien avulla. [6, s.131.] Assetin luonnissa valitaan sille sopiva type.
Data point	Data pointien avulla voidaan tuoda dataa datan lähteestä. Data pointit yhdistetään vastaavaan aspecttiin ja sen muuttujiin.
Data source	Datan lähde on laite, joka lähettää tietoja MindSphereen [6, s. 131].
Fleet Manager	Fleet Manager on visuaalinen työkalu, jolla voidaan tarkastella olemassa olevia asetteja ja niiden sisältämää dataa [6, s.131]. Dataa voidaan tarkastella mm. aikajaksoisina historiakuvaajina.
Type	Typet ovat MindSpheressä käytettäviä mallipohjia, jotka mahdollistavat aspectin ja assetin välisen yhteyden. [6, s. 62].

3.2 Aspectin ja tyypin luonti

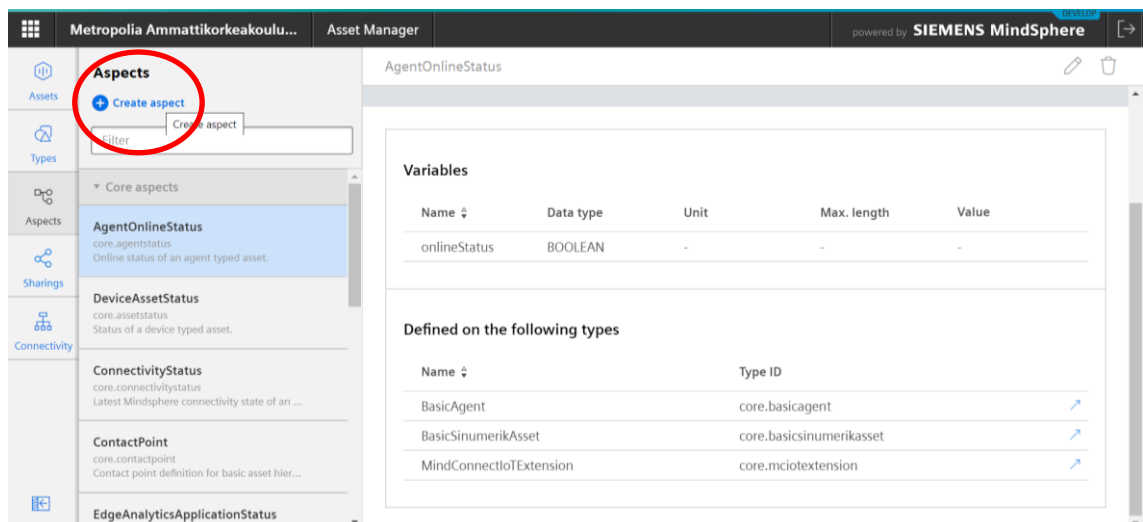
3.2.1 Aspectin luonti

- Aluksi valitaan MindSpheren aloitusnäkömstä Asset Manager.
- Asset Manager -sovelluksen aloitusnäköm avaa Assets-välilehden, joka näyttää tilille luodut assetit, joita esimerkkitapauksessa ei olla vielä luotu. Valitaan vasemmalta Aspects-välilehti (kuva 42).



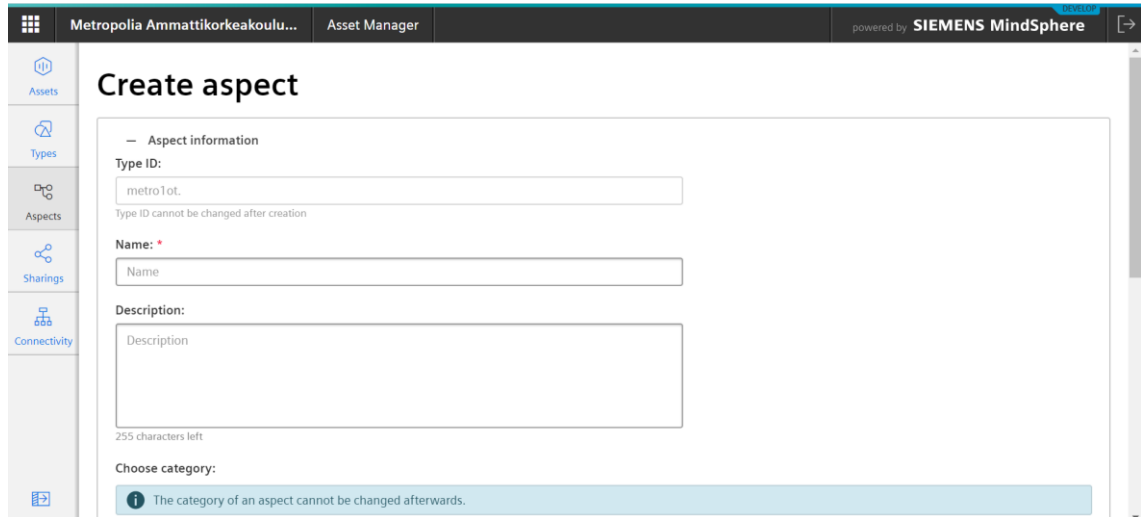
Kuva 42. Asset Manager-sovelluksen aloitusnäköm.

- Tämän jälkeen luodaan uusi aspect valitsemalla Create aspect (kuva 43).



Kuva 43. Aspects-välilehti. Create aspect -valinta ympyröitynä punaisella.

- Nimetään luotava aspect sopivalla nimellä kirjoittamalla Name-kohtaan nimi. Description-kohtaan voidaan kertoa tarkemmin mitä aspect tekee. Esimerkissä aspect nimetään nimellä Temperature, koska projektissa halutaan mitata lämpötilaa pariinkin eri otteeseen (kuva 44).



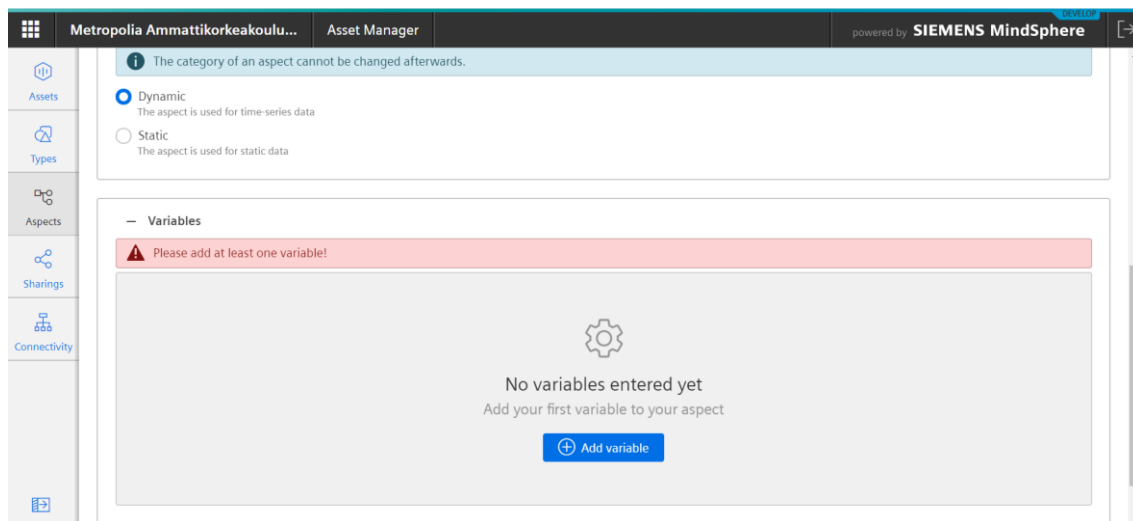
The screenshot shows the 'Create aspect' form in the Siemens MindSphere Asset Manager. The form is titled 'Create aspect' and is part of the 'Aspect information' section. It contains the following fields:

- Type ID:** metro1ot. (Note: Type ID cannot be changed after creation)
- Name:** (Required field, currently empty)
- Description:** (Text area, currently empty, with 255 characters left)

At the bottom, there is a 'Choose category' section with a warning message: 'The category of an aspect cannot be changed afterwards.'

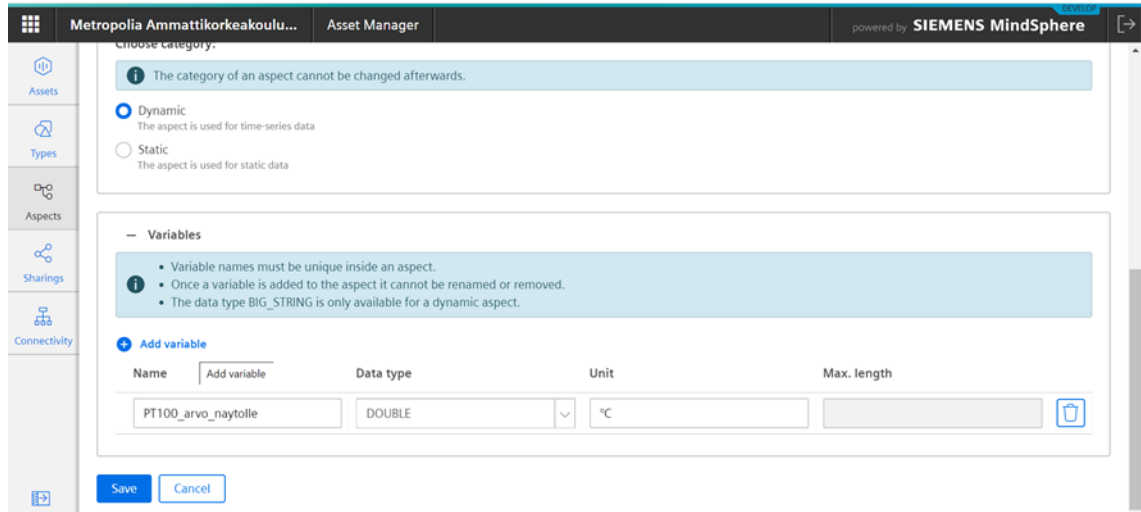
Kuva 44. Aspectin luonti-ikkuna.

- Alaspäin rullaamalla hiirellä samassa ikkunassa voidaan valita, onko aspect dynaaminen vai staattinen. Tämä riippuu aspectiin määriteltävien muuttujien tyy-pistä. Analogiset muuttujat määritellään dynaamisiksi ja digitaaliset muuttujat staattisiksi. Esimerkkitapauksessa valitaan dynaaminen, koska Temperature-aspectin muuttujat ovat analogisia. HUOM! MindSpheren koodissa oli vika ainakin tammikuun 2020 versiossa, joten myös digitaalisia muuttujia sisältään aspectin kannattaa valita dynaamisiksi.
- Seuraavaksi lisätään aspectin muuttujat valitsemalla Add variable (kuva 45).



Kuva 45. Dynamic ja Static -valinta sekä muuttujien lisääminen

- Lisätään muuttujien tiedot aspectiin kohtiin Name, Data type ja Unit. Esimerkitapauksessa muuttujat nimetään TIA Portalissa olevan PLC-projektin muuttujien mukaisesti ja datatyyppi PLC-projektin muuttujien datatyyppin mukaisesti. Tyyppiä Double ei suoraan PLC-projektin muuttujissa näy, mutta kuvassa 47 näkyvässä TIA Portalin tag-listan Address-kohdassa näkyy. Esimerkiksi PT100_T_naytolle-muuttujan osoite on MD4, jonka ensimmäinen kirjain kertoo muuttujan olevan muistimuuttuja, toinen kirjain kertoo muuttujan olevan Double-tyyppiä ja numeroosa kertoo muuttujan sijaitsevan bittimuistipaikassa 4. Unit-kohtaan tarvitaan jokin merkki, jotta muuttujat pystytään hyväksymään. Lämpötilaa ilmaistakseen tässä on käytetty merkkejä ° ja C (kuva 46).
 - o Muuttujien datatyyppivaihtoehtoja on seitsemän: boolean, int, long, double, string, big_string ja timestamp. Getting Connected to MindSphere -ohjeen sivuilta 72–78 löytyy tietoja S7-protokollan konfiguroinnista sekä käytettävistä datatyypeistä.
- Tässä ei ole pakko käyttää PLC-muuttujien nimiä, sillä konfigurointi PLC-logiikan ja MindSpheren välillä tapahtuu TIA Portalin muuttujan osoitteen välityksellä, mutta selkeyden takia esimerkitapauksessa käytetään samoja nimiä.
- Valitsemalla Add variable saadaan halutessaan uusi tyhjä rivi, johon voidaan lisätä toinen muuttuja.



Kuva 46. Muuttujien luonti aspectin sisällä.

PLC tags									
	Name	Tag table	Data type	Address	Retain	Acces...	Writa...	Visibl...	
1	PT100_T_arvo	Default tag table	Int	%IW4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	PT100_T_muutos	Default tag table	Real	%MD0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	PT100_T_naytolle	Default tag table	Real	%MD4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	EE800_T_arvo	Default tag table	Int	%IW6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	EE800_T_muutos	Default tag table	Real	%MD8	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	EE800_T_naytolle	Default tag table	Real	%MD12	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	EE800_CO_arvo	Default tag table	Int	%IW8	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	EE800_CO_muutos	Default tag table	Real	%MD16	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	EE800_CO_naytolle	Default tag table	Real	%MD20	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	Capasitive_sensor	Default tag table	Bool	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	LED	Default tag table	Bool	%Q1.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12	<Add new>				<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Kuva 47. PLC-projektin muuttujat.

- Lopuksi kun haluttu määrä muuttujia on luotu, painetaan Save. HUOM! Muuttujia voidaan lisätä tämänkin jälkeen luotuun aspectiin, mutta jo luotuja muuttujia ei voida poistaa. Mikäli huomaat tehneesi virheen muuttujien suhteen, joudut valittavasti luomaan uuden aspectin ja tarvittaessa poistamaan edellisen (kuva 48).

Metropolia Ammattikorkeakoulu... Asset Manager powered by SIEMENS MindSphere

Dynamic
The aspect is used for time-series data

Static
The aspect is used for static data

— Variables

- Variable names must be unique inside an aspect.
- Once a variable is added to the aspect it cannot be renamed or removed.
- The data type BIG_STRING is only available for a dynamic aspect.

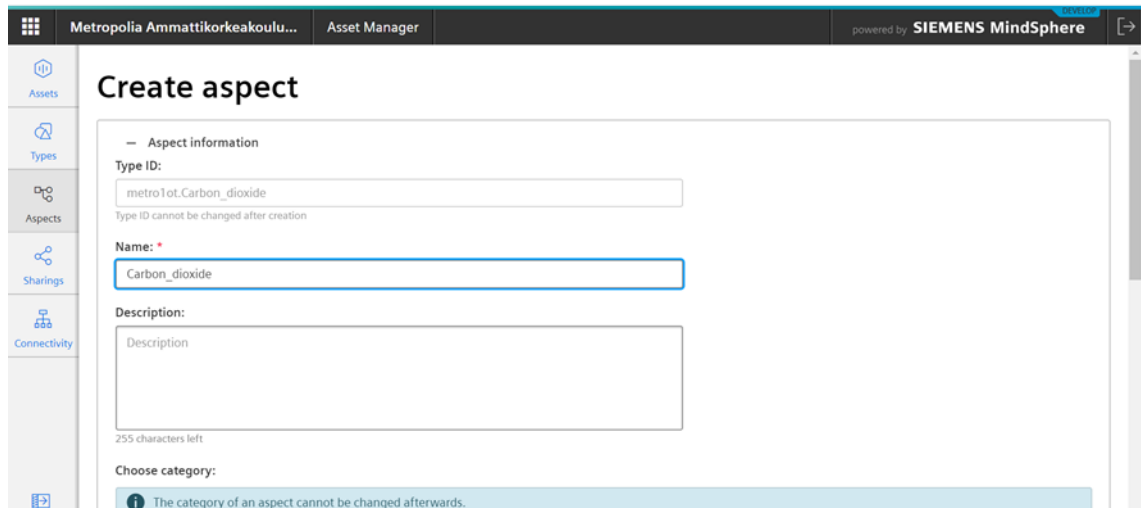
+ Add variable

Name	Data type	Unit	Max. length
PT100_T_naytolle	DOUBLE	°C	
EE800_T_naytolle	DOUBLE	°C	

Save Cancel

Kuva 48. Aspectin tallennus.

- Esimerkkiprojektissa luodaan vielä kaksi aspectia: Carbon_dioxide, johon luodaan yksi muuttuja ja Proximity_IOs, johon luodaan myös yksi muuttuja. Jälkimmäisessä aspectissa luodaan boolean-tyyppin muuttuja, joka on joko tilassa True tai False. Tässäkin valitaan dynaaminen kategoria aiemmin mainitun ongelman takia (kuvat 49 ja 50). Boolean-tyyppin muuttujan yksikkö jätetään tyhjäksi, mutta ohjelma haluaa muuttujalle jonkun yksikön. Tähän käy esimerkiksi välilyönti, sillä muuten aspectia ei voida tallentaa.



Metropolia Ammattikorkeakoulu... Asset Manager powered by SIEMENS MindSphere

Create aspect

— Aspect information

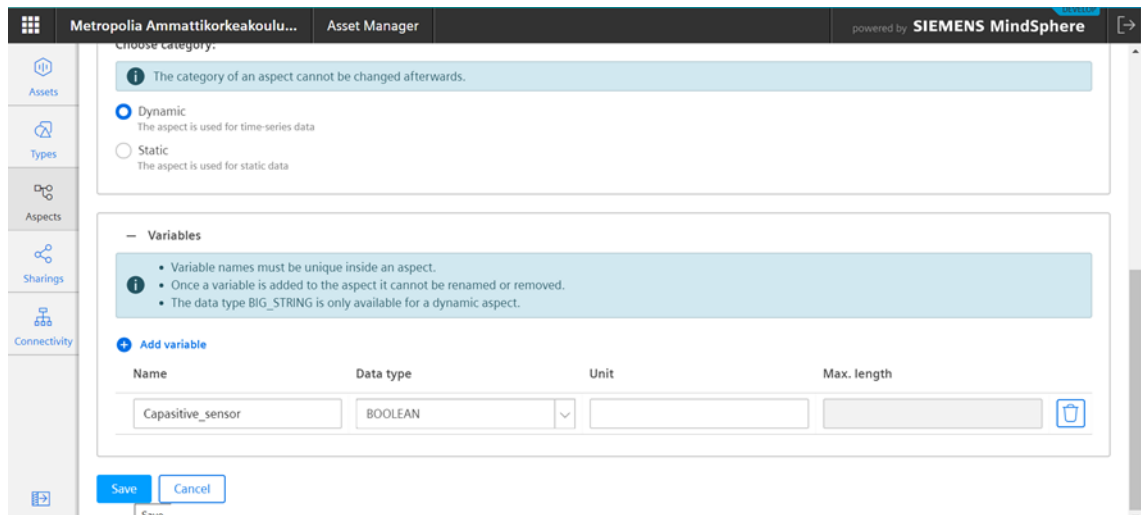
Type ID:
metro1ot.Carbon_dioxide
Type ID cannot be changed after creation

Name: *
Carbon_dioxide

Description:
Description
255 characters left

Choose category:
The category of an aspect cannot be changed afterwards.

Kuva 49. Toisen aspectin luonti.



Choose category:
The category of an aspect cannot be changed afterwards.

Dynamic
The aspect is used for time-series data

Static
The aspect is used for static data

— Variables

- Variable names must be unique inside an aspect.
- Once a variable is added to the aspect it cannot be renamed or removed.
- The data type BIG_STRING is only available for a dynamic aspect.

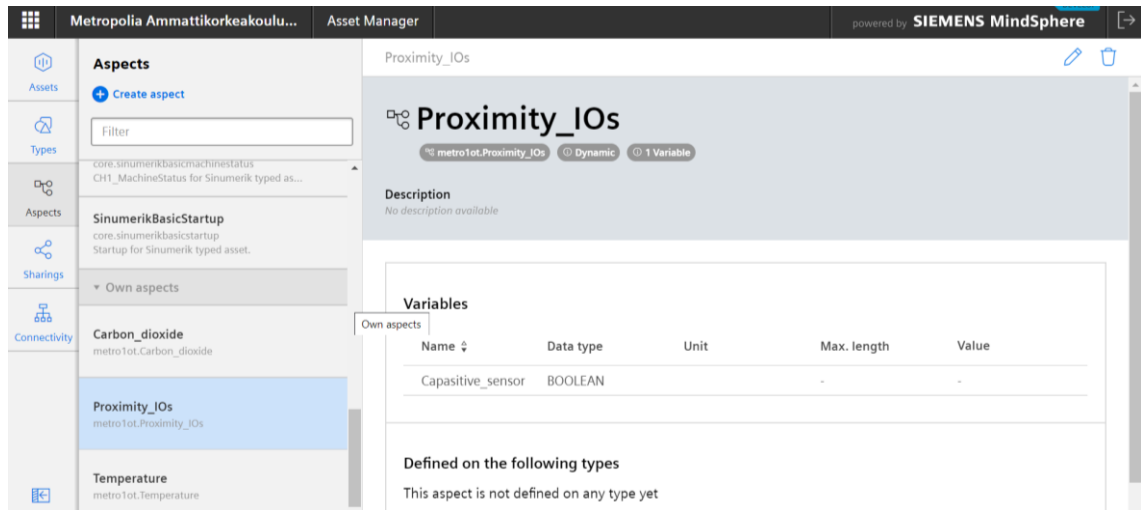
+ Add variable

Name	Data type	Unit	Max. length
Capacitive_sensor	BOOLEAN		

Save Cancel

Kuva 50. Boolean-tyyppisen muuttujan luonti kolmanteen aspectiin.

- Esimerkkiprojektissa luodut kolme aspectia näkyvät Aspects-välilehdet Own aspects -alavalikon alla (kuva 51).

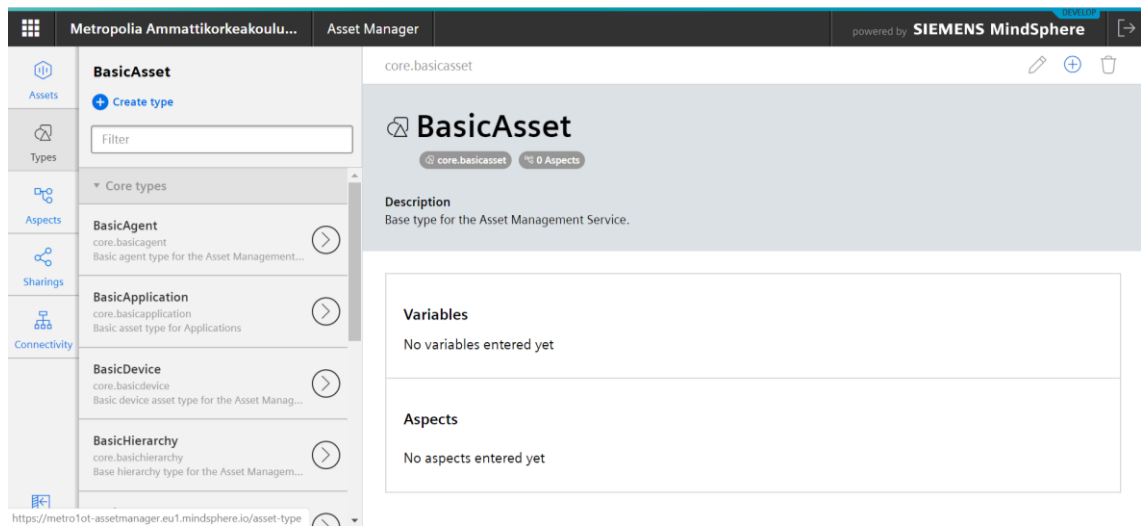


Kuva 51. Kolme luotua aspectia Aspects-välilehdellä.

3.2.2 Typen luonti

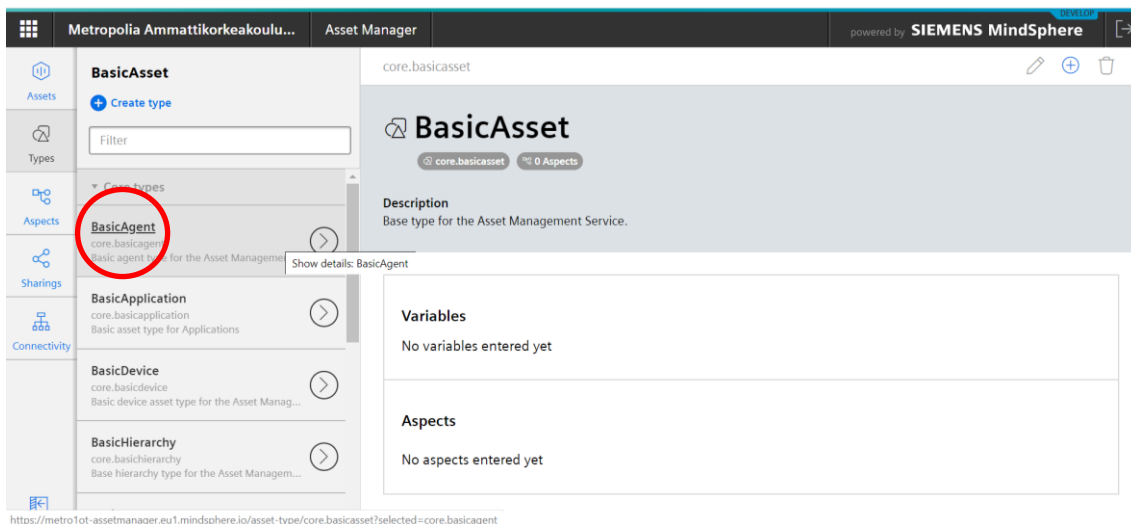
Aspectien luonnin jälkeen luodaan tyyppi, johon luodut aspectit linkitetään.

- Valitaan vasemmalta Types-välilehti (kuva 52).



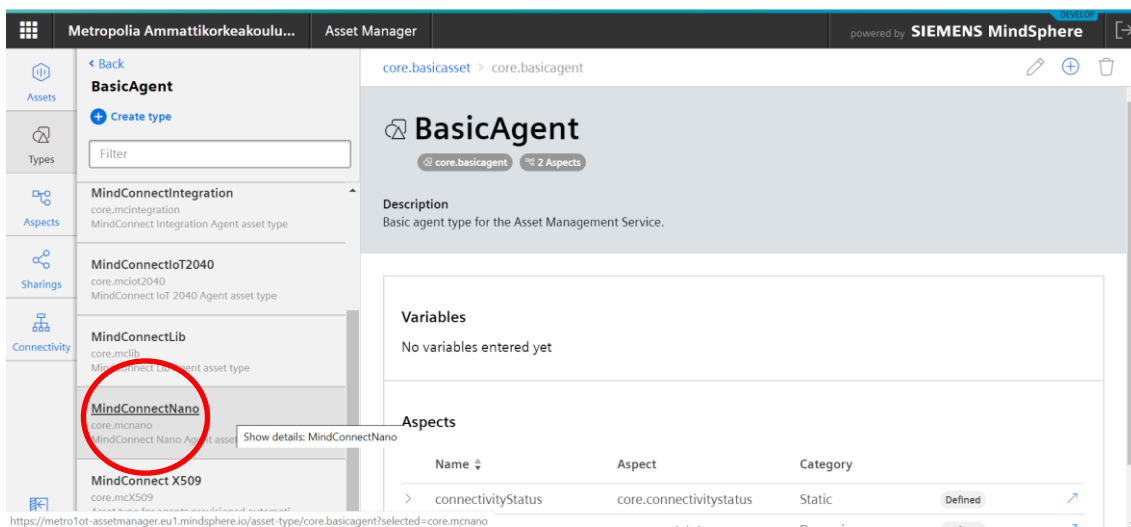
Kuva 52. Types-välilehti

- Valitaan BasicAgent-valikko, joka ympyröity kuvassa 53 punaisella.



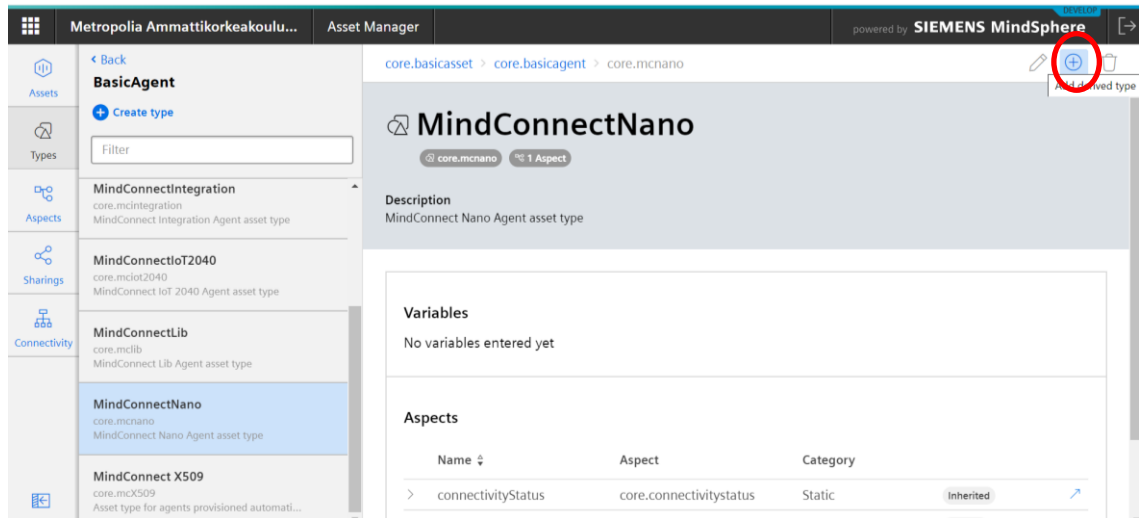
Kuva 53. Types-välilehdeltä valitaan BasicAgent.

- Valitaan BasicAgent-valikosta projektissa käytettävä MindConnect-laite eli joko MindConnect Nano tai MindConnect IoT2040. Esimerkissä valitaan MindConnect Nano, joka on kuvassa 54 ympyröity punaisella.



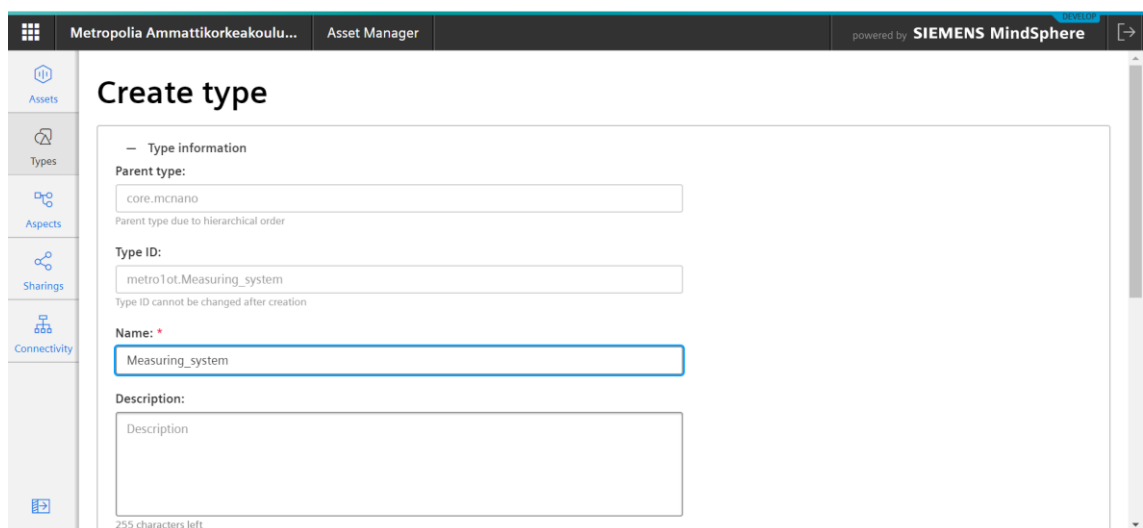
Kuva 54. BasicAgent-valikko.

- MindConnect Nano -välilehdeltä valitaan Add derived type, joka on kuvassa 55 ympyröity punaisella.



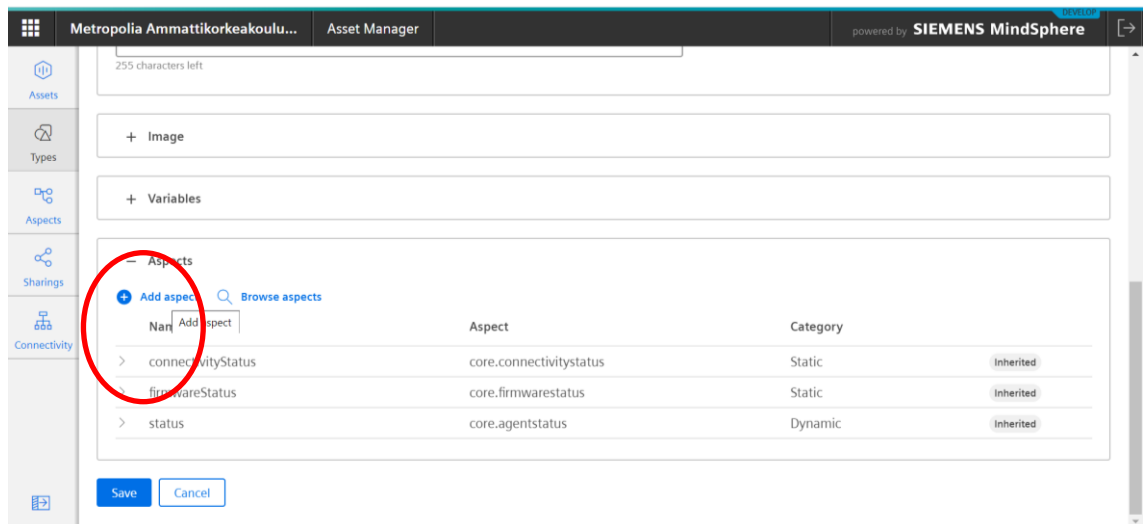
Kuva 55. MindConnect Nano -välilehti.

- Luotava tyyppi nimetään kohdassa Name sopivalla tavalla. Esimerkkiprojektissa käytetään nimitystä Measuring_system (kuva 56).



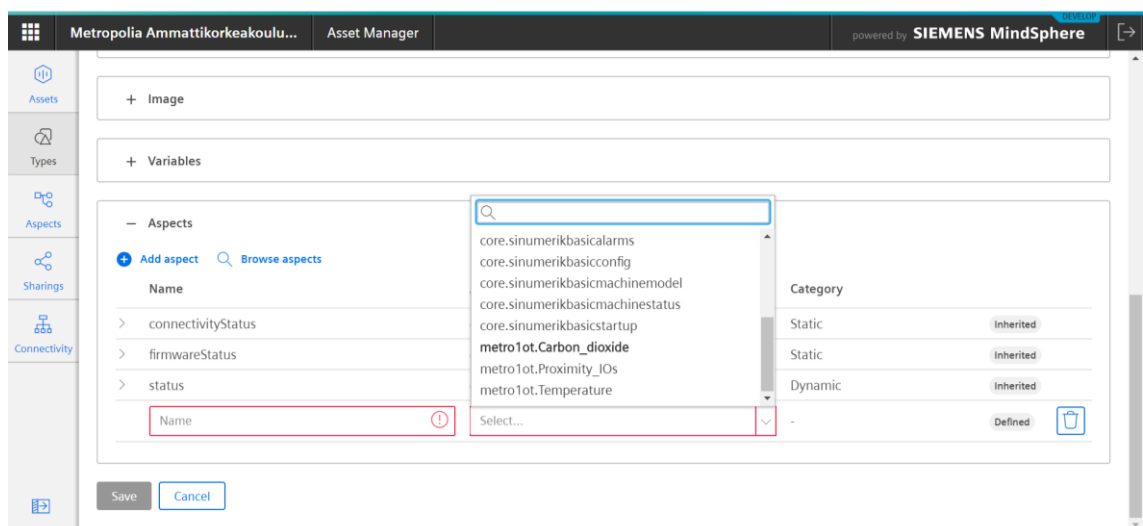
Kuva 56. Tyypin luonti-ikkuna.

- Rullataan hiirellä alaspäin ja painetaan kohdassa Aspects painiketta Add aspect, joka on ympyröity kuvassa 57.



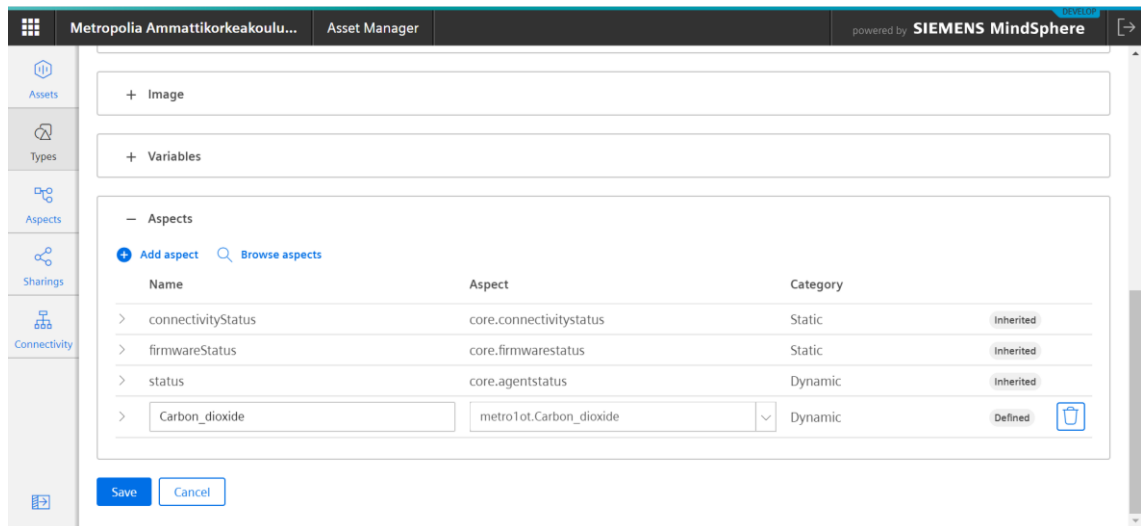
Kuva 57. Aspectien lisäys tyyppiin.

- Keskellä olevasta valintalaatikosta etsitään joku luoduista aspecteista. Kuvassa 58 etsitään Carbon_dioxide -aspect.



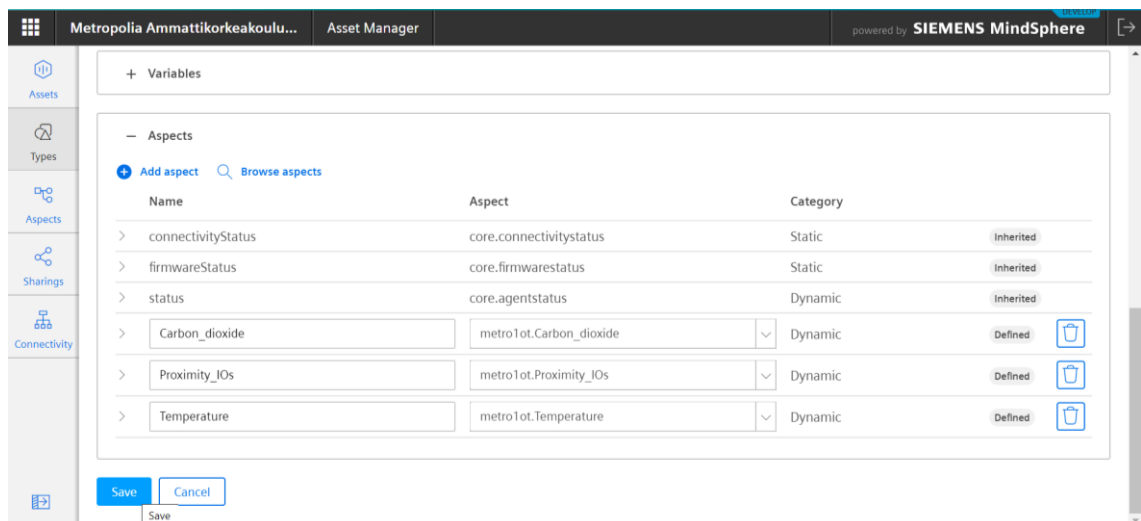
Kuva 58. Alavalikosta etsitään haluttu aspect.

- Valinnan jälkeen aspect nimeää itsensä oletusarvoisella nimellä, kuten kuvassa 59.



Kuva 59. Löydetty aspect.

- Kuvassa 60 näkyy esimerkkitapauksen kolme aspectia luotavan tyyppin sisällä. Lopuksi painetaan Save. Tyyppi on luotu.

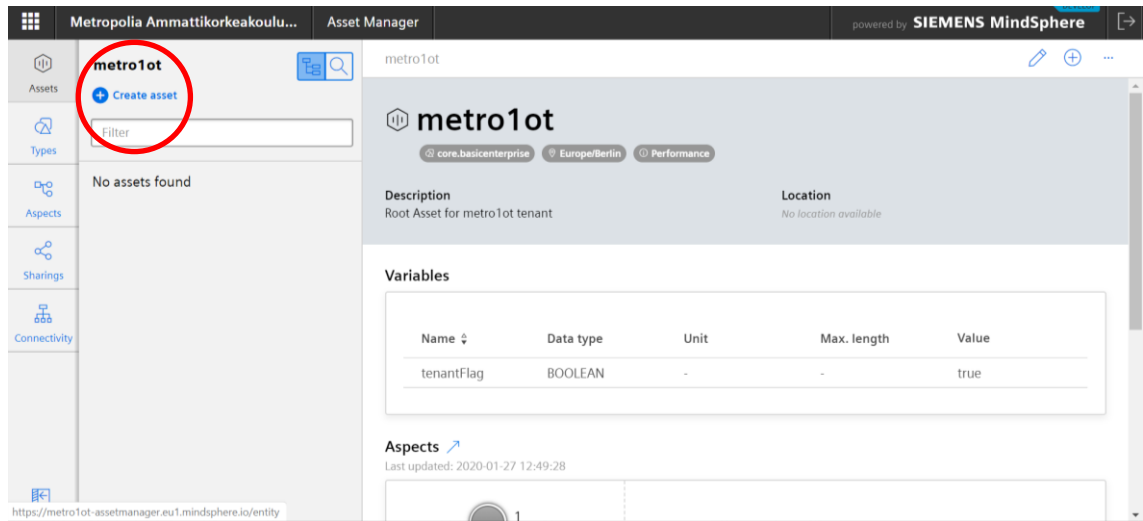


Kuva 60. Esimerkkitaupauksen kolme aspectia.

3.3 Assetin luonti

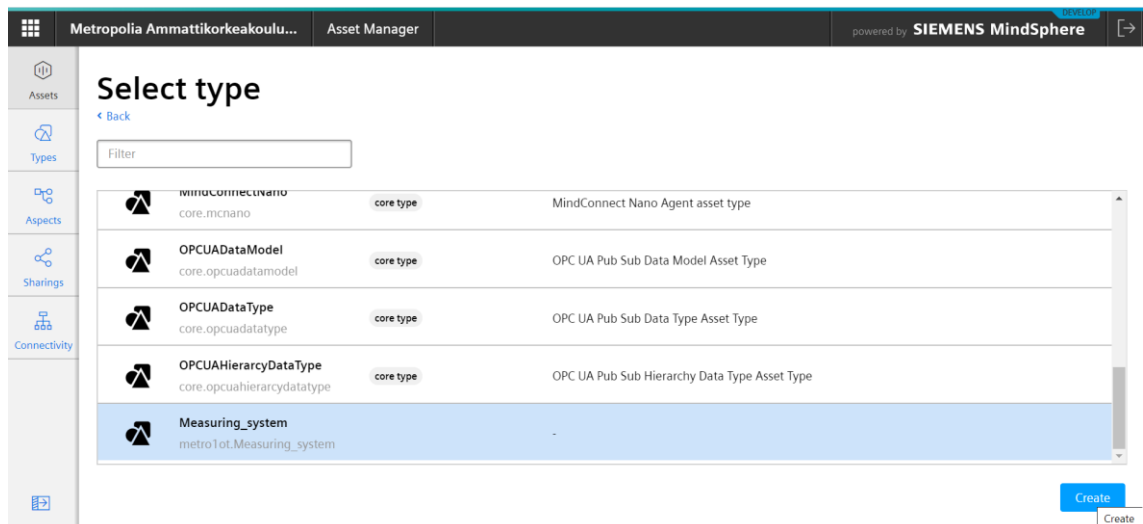
Kun tyyppi on luotu, johon ollaan linkitetty luodut aspectit, voidaan siirtyä assetin luontiin.

- Asset-välilehdeltä valitaan Create asset (kuva 61).



Kuva 61. Asset-välilehti. Create asset -valinta ympyröity punaisella.

- Selecting type -valikosta etsitään luotu tyyppi. Esimerkitapauksessa Measuring_system. Valitaan se ja painetaan Create (kuva 62).



Kuva 62. Tyypin valintaikkuna.

- Nimetään asset kohdassa Name sopivalla tavalla. Esimerkkiprojektissa nimeäminen on tehty nimellä Potato Nano (kuva 63).

Metropolia Ammattikorkeakoulu... Asset Manager powered by SIEMENS MindSphere

Add asset

Assets

Types

Aspects

Sharings

Connectivity

— General

Type ID: metro1ot.Measuring_system
Selected type of asset cannot be changed

Name: * Potato Nano

Description: Description
255 characters left

+ Location

Kuva 63. Asetin nimeäminen.

- Lopuksi tallennetaan luotava asset painamalla Save (kuva 64).

Metropolia Ammattikorkeakoulu... Asset Manager powered by SIEMENS MindSphere

+ Variables

— Aspects

Name	Aspect	Category
> Carbon_dioxide	metro1ot.Carbon_dioxide	Dynamic
> connectivityStatus	core.connectivitystatus	Static
> firmwareStatus	core.firmwarestatus	Static
> Proximity_IOs	metro1ot.Proximity_IOs	Dynamic
> status	core.agentstatus	Dynamic
> Temperature	metro1ot.Temperature	Dynamic

* Required input field

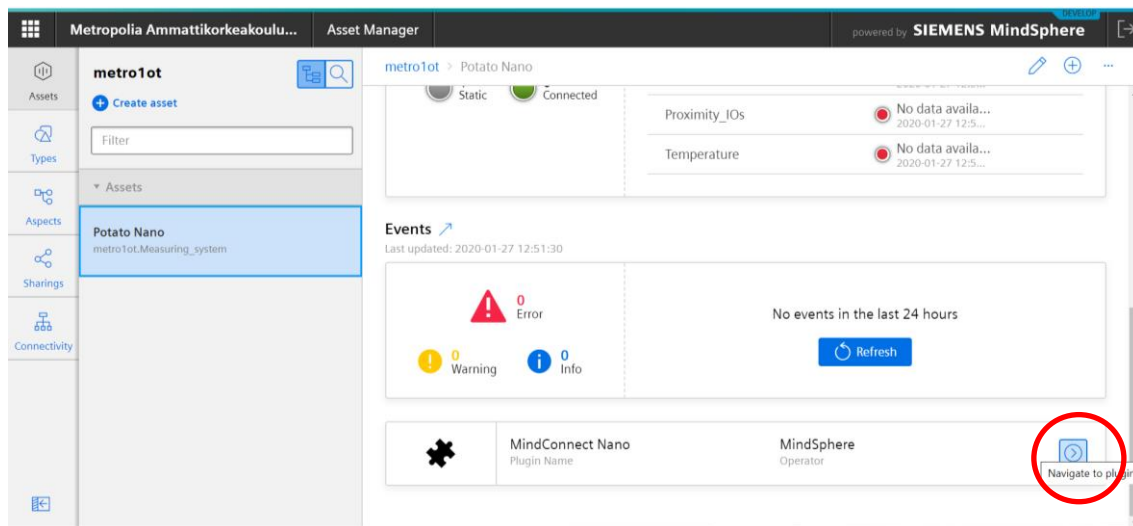
Save Cancel

Kuva 64. Asetin tallennus.

3.4 MindSphereen kytkeytyminen

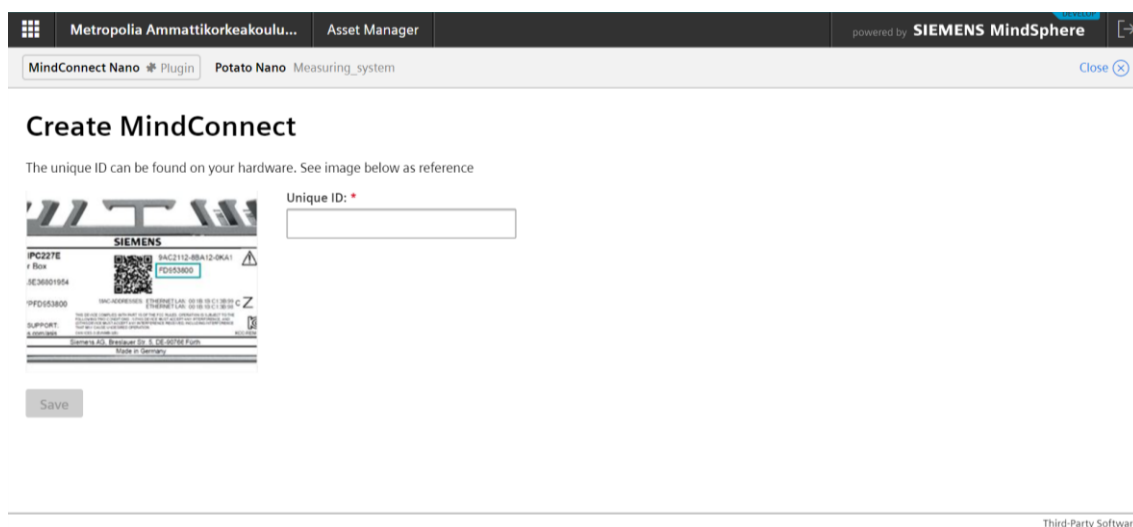
3.4.1 MindSpheren reititykset

- Asetin luomisen jälkeen valitaan luotu asset Asset-välilehdeltä ja etsitään MindConnect-laite assetin ikkunasta. Esimerkitapauksessa MindConnect Nano (kuva 65).



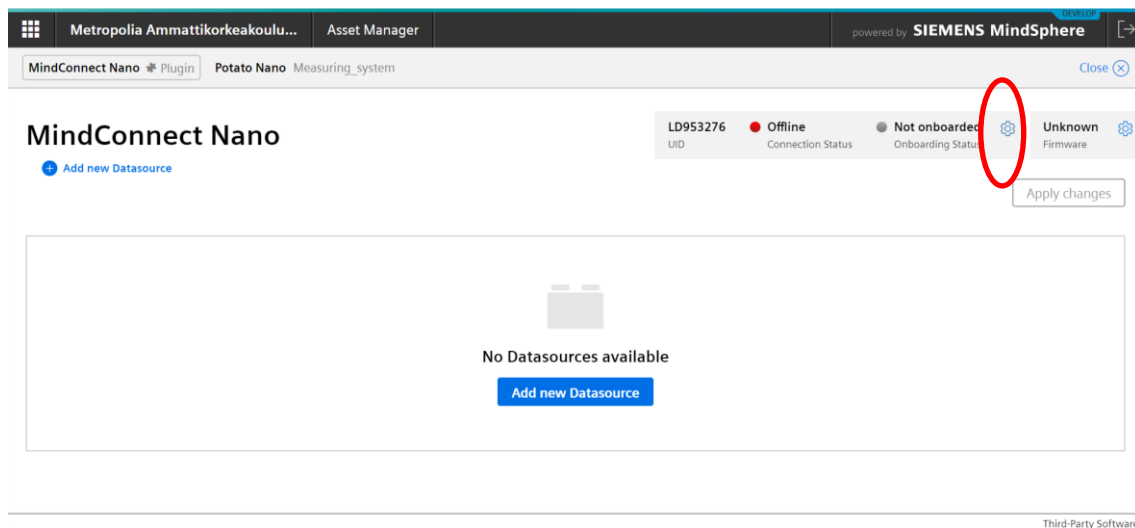
Kuva 65. Luodun assetin ikkuna. MindConnect Nano -laitteen valinta ympyröity punaisella.

- Seuraavassa ikkunassa kysytään MindConnect-laitteen yksilöllistä ID-tunnusta, joka löytyy fyysisen MindConnect-laitteen tunnistetarrasta QR-koodin vierestä kuvassa 66 mukaisesti. Koodissa on kahdeksan kirjainta.



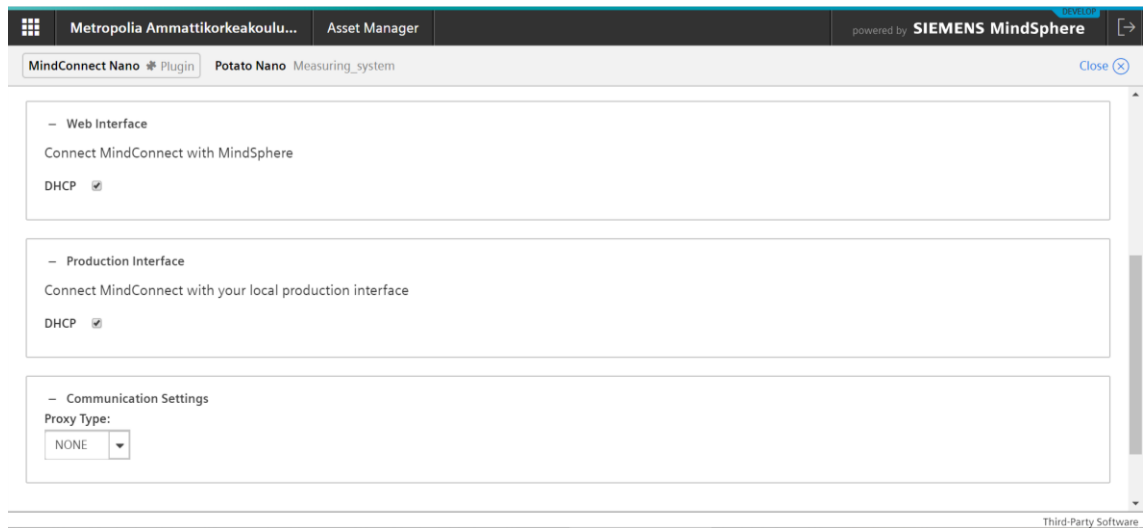
Kuva 66. MindConnect-yhteyden luontiin tarvittava yksilöllinen laitteen ID-tunnus.

- Seuraavaksi määritetään MindConnect Nanon reititykset painamalla kuvassa 67 näkyvää punaisella ympyrällä merkittyä ratasta.

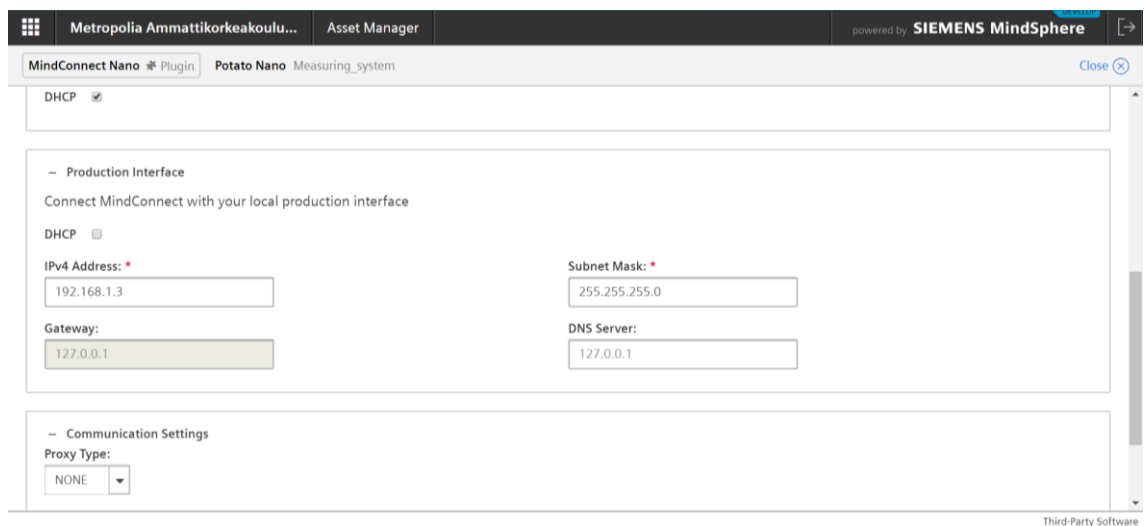


Kuva 67. Laitteen yhteysikkuna.

- Avautuvasta ikkunasta voidaan määrittää MindConnect-laitteen Ethernet-asetukset. Web Interface -kohdassa voidaan määrittää millä osoitteella MindConnect-laitte löydetään Ethernet-verkosta. Esimerkitapauksessa annetaan käytettävissä olevan DHCP-palvelimen etsiä sopiva IP-osoite, kuten mainitaan aiemmin Esivalmistelu-luvun kuvassa 8 (kuva 68).
- Production Interface -kohdassa voidaan määrittää IP-osoite, jolla MindConnect-laitte on yhteydessä tuotannon verkkoon. Esimerkitapauksessa, kuten Esivalmistelu-kappaleen kuvassa 8 näytetään, valitaan osoitteeksi 192.168.1.3, joka on samassa verkossa kuin PLC-logiikka (kuva 69). Lisäksi aliverkon maski on asetettava samaksi kuin PLC-logiikan vastaava. Esimerkin tapauksessa 255.255.255.0. Kuvassa 69 näkyvät loput valintaruudut on merkitty harmaalla tekstillä 127.0.0.1. Ne voi antaa olla sellaisinaan.



Kuva 68. Laitteen Ethernet-asetusikkuna.



Kuva 69. Esimerkkitapauksen IP-osoitteen määrittäminen tuotantoverkossa.

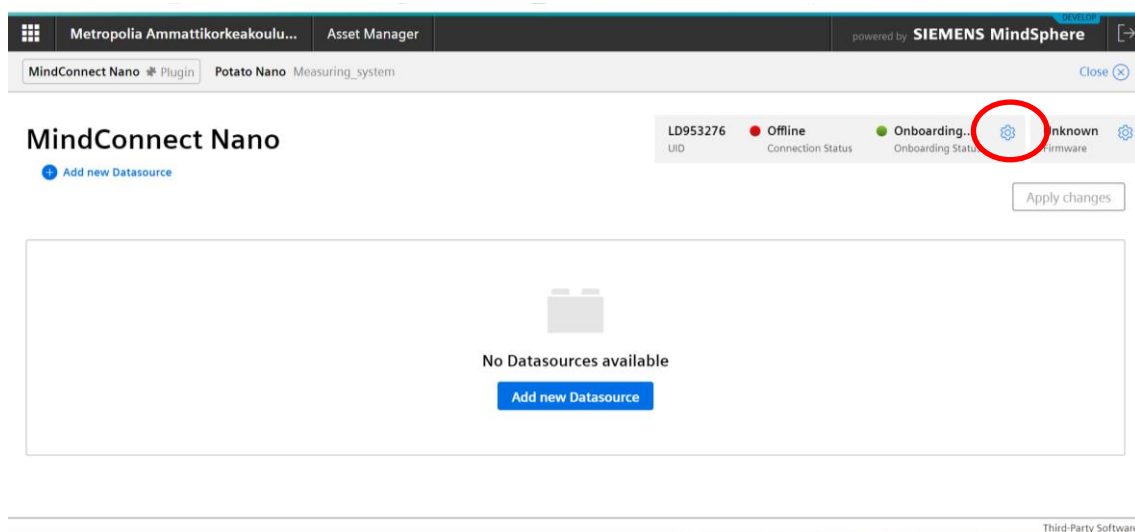
- Lopuksi tallennetaan asetukset painamalla Save. Tämä vielä halutaan todentaa varmistusikkunalla, johon painetaan Yes (kuva 70).

The screenshot shows the configuration page for a MindConnect Nano device. The interface includes a header with the application name and a 'powered by SIEMENS MindSphere' logo. Below the header, there are tabs for 'MindConnect Nano' and 'Potato Nano'. The main content area is titled 'DHCP' and contains several input fields: 'IPv4 Address' (192.168.1.3), 'Subnet Mask' (255.255.255.0), 'Gateway' (127.0.0.1), and 'DNS Server' (127.0.0.1). There is also a 'Communication Settings' section with a 'Proxy Type' dropdown menu set to 'NONE'. At the bottom, there are 'Save' and 'Cancel' buttons and a note '* required input field'.

Kuva 70. Asetusten tallennus.

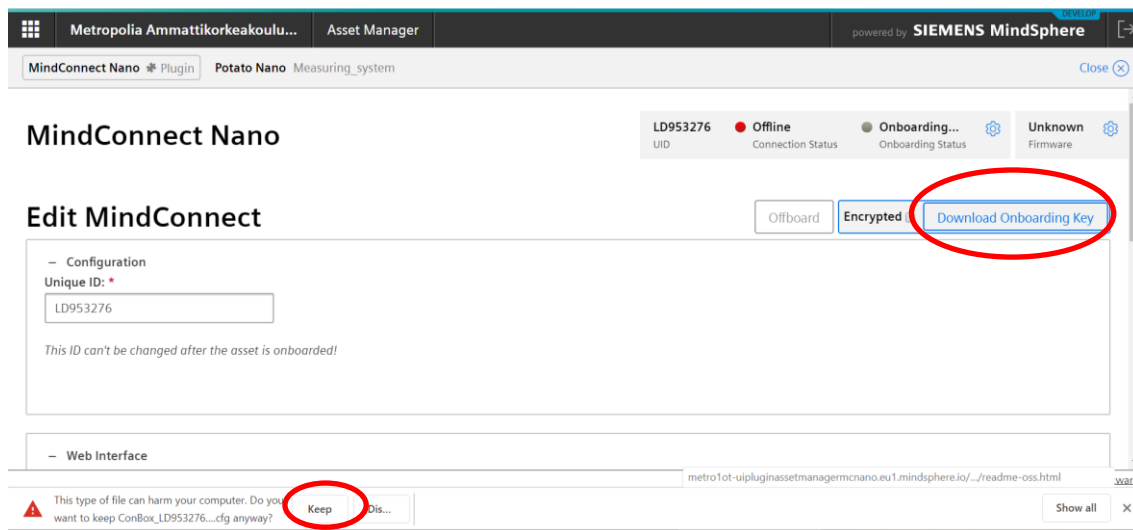
3.4.2 MindConnect-laitteen Onboarding

- Seuraavaksi asetetaan USB-tikku tietokoneeseen, jossa MindSphere on auki selaimella.
- Painetaan uudestaan kuvassa 71 punaisella ympyrällä merkittyä ratasta.



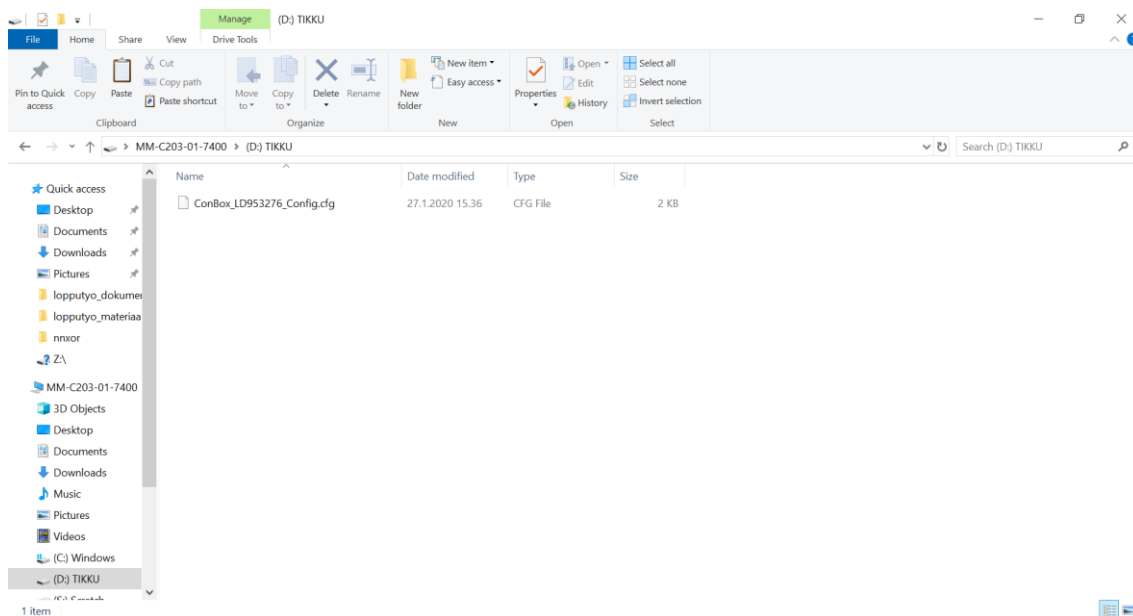
Kuva 71. MindConnect Nano -yhteysikkuna.

- Painetaan painiketta Download Onboarding Key. Selain kysyy, että tämälntapainen tiedosto voi vahingoittaa konetta. Painetaan vaihtoehtoa Keep (kuva 72).



Kuva 72. MindConnect Nanon asetussivuna, josta voidaan ladata asetukset fyysiselle laitteelle.

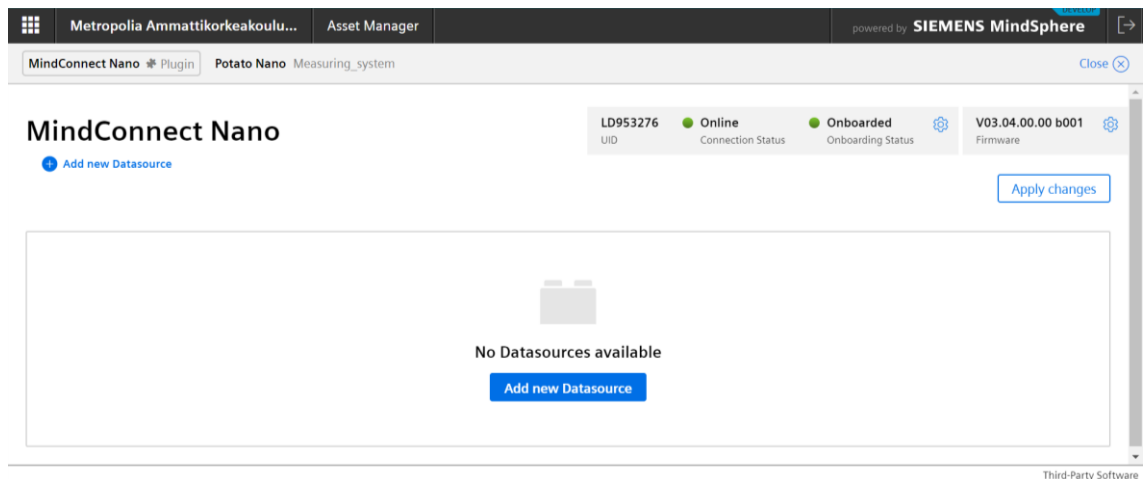
- Kopioidaan tai siirretään ladattu tiedosto tyhjälle USB-tikulle. Tiedoston on oltava nimetty *ConBox_XXXXXXXX_Config.cfg*, x-kirjainten ollessa yksilöllinen ID-koodi. Nimessä ei saa olla ylimääräisiä merkkejä, kuten esimerkiksi uudelleen kopiointia ilmaisevaa (1)-merkkiä (kuva 73).



Kuva 73. Ladatun konfigurointitiedoston siirto USB-tikulle.

- Siirretään USB-tikku tietokoneesta MindConnect-laitteen USB-porttiin. Esimerkkitapauksessa USB-tikku siirretään MindConnect Nano -laitteen oletus-USB-porttiin X63, kuten Esivalmistelut-luvun kuvassa 5 näkyy.

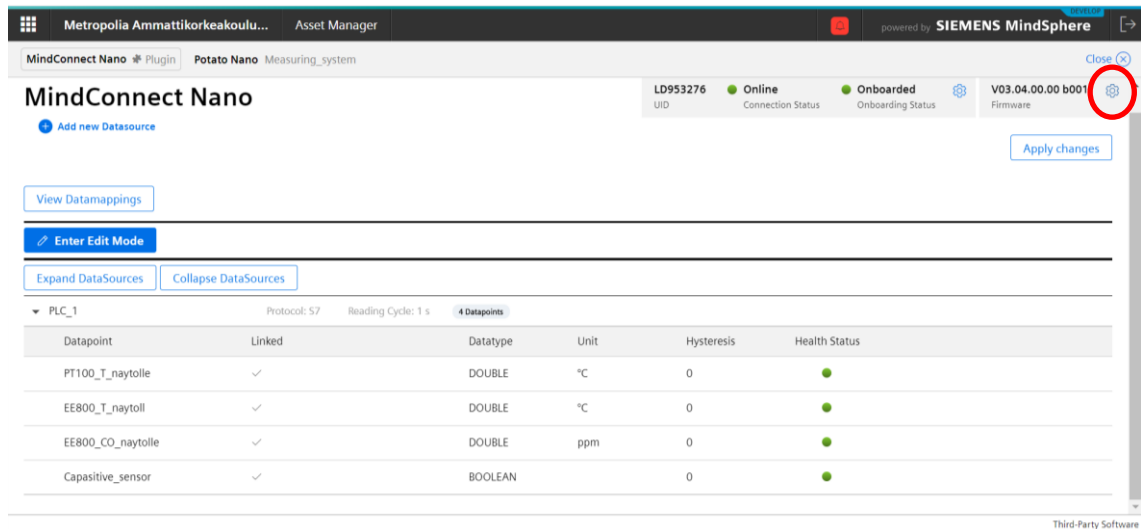
- Esimerkkitapauksessa MindConnect Nano -laitteen valot alkavat vilkkua. LED-valojen toiminnoista löytyy ohjeet Getting Connected to Siemens -ohjeesta sivuilta 26–32. Kun valojen vilkkuminen lakkaa ja vihreät LEDit palavat LED-valoissa PC ON ja RUN, niin yhteys MindSpheren ja MindConnect Nanon välille on syntynyt. Tämän jälkeen USB-tikun voi poistaa laitteen USB-portista.
- Päivittämällä selaimen MindConnect Nano -yhteysikkunassa, voidaan nähdä kuvassa 74 yhteyden syntyneen, kun yhteysikkunan Online- ja Onboarding-valot ovat vihreinä.



Kuva 74. MindConnect Nano-yhteysikkuna näyttää yhteyden olevan online-tilassa sekä MindConnect-laitteen ja MindSpheren olevan yhteydessä.

3.4.3 MindConnect-laitteen alustaminen ja uuden firmwaren asennus

HUOM! Tämä kappale vain tarvittaessa. MindConnect-laitteeseen voidaan vaihtaa laiteohjelmiston firmware suoraan MindSpherestä painamalla kuvassa 75 näkyvää punaisella ympyrällä olevaa ratasta MindConnect laitteen ollessa online-tilassa sekä yhteydessä MindSphereen.



Kuva 75. Yhteysikkuna.

Mikäli laitteen yhdistämisessä online-tilaan tai MindSpheren on ongelmia, firmwaren asennus onnistuu myös ilman online-yhteyttä manuaalisesti, jotka voivat auttaa yhteysongelmiin.

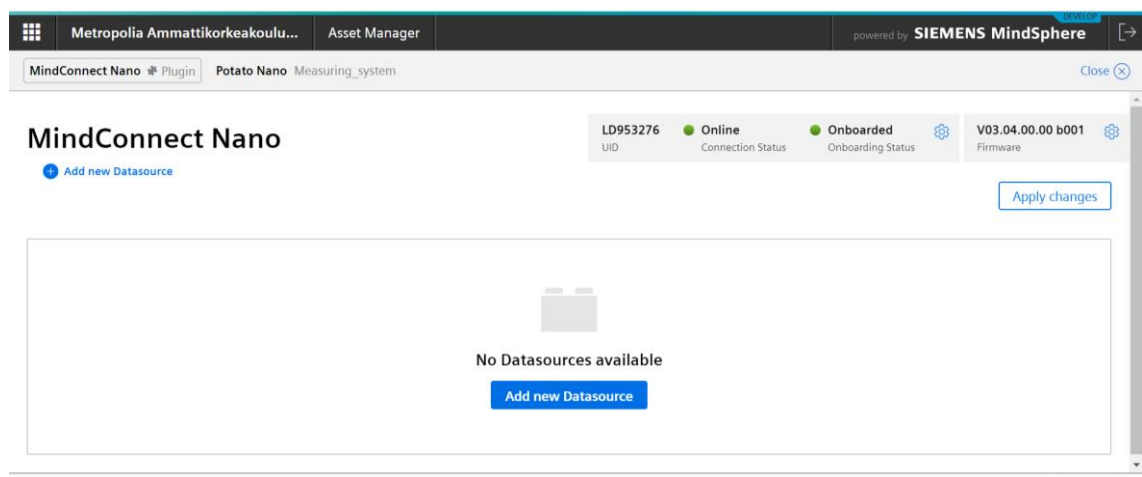
- Aluksi MindConnect-laite on asetettava tehdasasetukset, mikäli laite on ollut yhteydessä MindSpheren versioon kaksi. Mikäli haluat mennä seuraavan vaiheeseen, siirry viisi ranskalaista viivaa alaspäin. Tyhjälle muistitikulle on luotava tekstitiedosto (tähän käy esimerkiksi Notepad-tiedosto), jonka sisälle on kirjoitettava seuraava teksti:

```
{
  "Commands": [
    {
      "Cmd": "DoFactoryReset",
      "DeviceID": "_MindConnect_ID_"
    }
  ]
}
```

- Tiedostoon kohtaan `_MindConnect_ID_` on kirjoitettava laitteen yksilöllinen ID-tunnus lainausmerkkien väliin.
- Tekstitiedosto on tallennettava nimellä `Conbox_Commands.json`. Lopuksi tarkasta, että tikulla on vain tämä tiedosto json-muodossa.
- Aseta tikku MindConnect-laitteen USB-paikkaan. MindConnect Nano-laitteessa paikkaan X63.
- Kun valot ovat lakanneet vilkkumasta ja esimerkiksi MindConnect Nano -laitteessa palavat sekä vihreä ON-LED-valo sekä oranssi RUN-LED-valo, voidaan tikku irroittaa.
- Seuraavaksi joko tehdasasetusten asentamisen jälkeen tai ilman niitä voidaan ladata uusin firmware-versio. Linkki löytyy lähteistä, mikäli Internetistä ei suoraan löydy firmwarea MindConnect-laitteelle [8; 9].
- Puretaan ladattu tiedosto ja siirretään kansion sisällä ollut .tar-tiedosto tyhjälle muistitikulle.
- Asetetaan tikku MindConnect-laitteeseen ja annetaan firmwaren asentua.
- Esimerkiksi MindConnect Nano -laitteella firmwaren asennuksessa kestää noin minuutin asentua. Asennuksen lopuksi palavat vihreänä ON-LED-valo sekä oranssina ERROR-LED-valo sen johdosta, että laite ei ole MindSphereen yhteydessä. Tällöin MindSphere-yhteys voidaan ottaa käyttöön suorittamalla Onboarding-prosessi uudestaan.
- Voit poistaa tikun MindConnect-laitteesta.

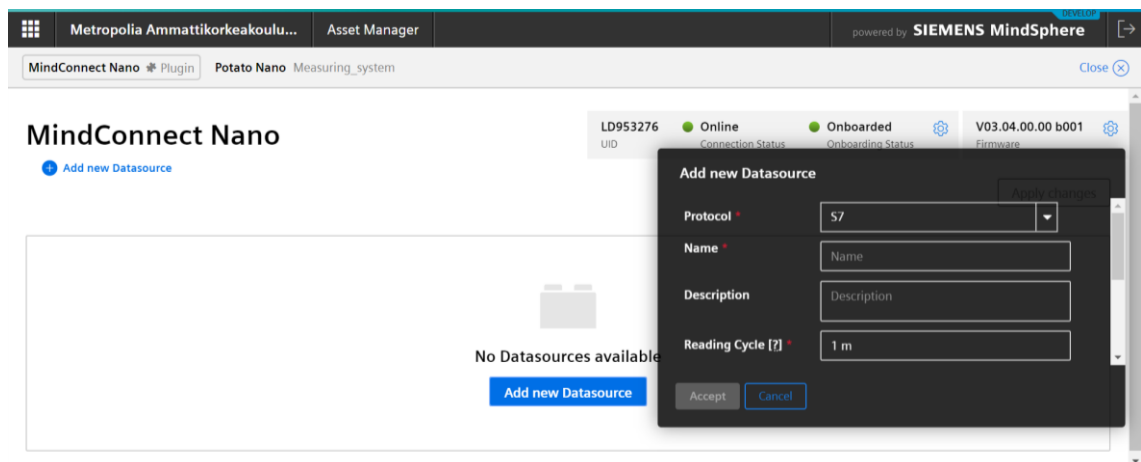
3.5 Datasourcen ja data pointien määrittäminen

- Painetaan MindConnect Nano -yhteysikkunasta Add new datasource (kuva 76).

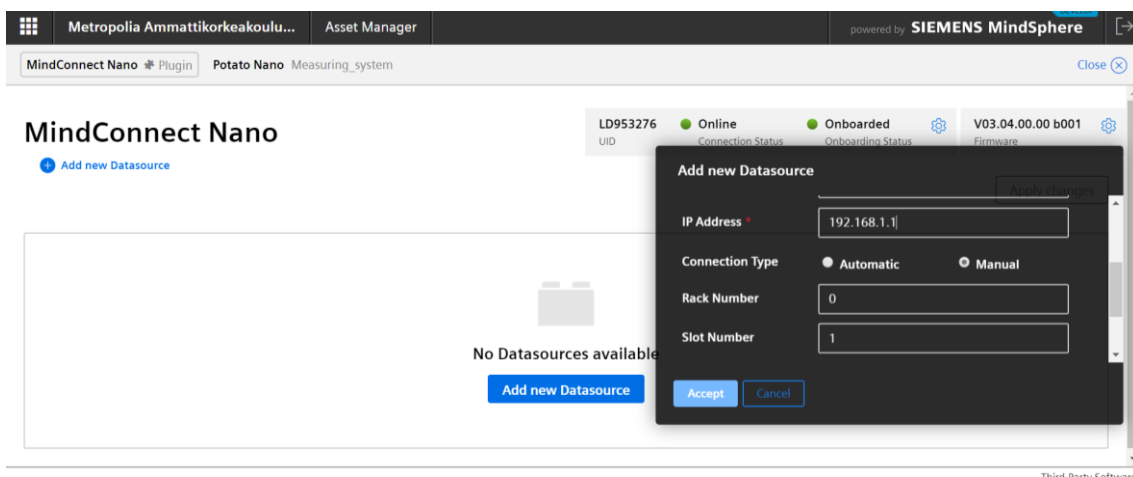


Kuva 76. Uuden datan lähteen luonti.

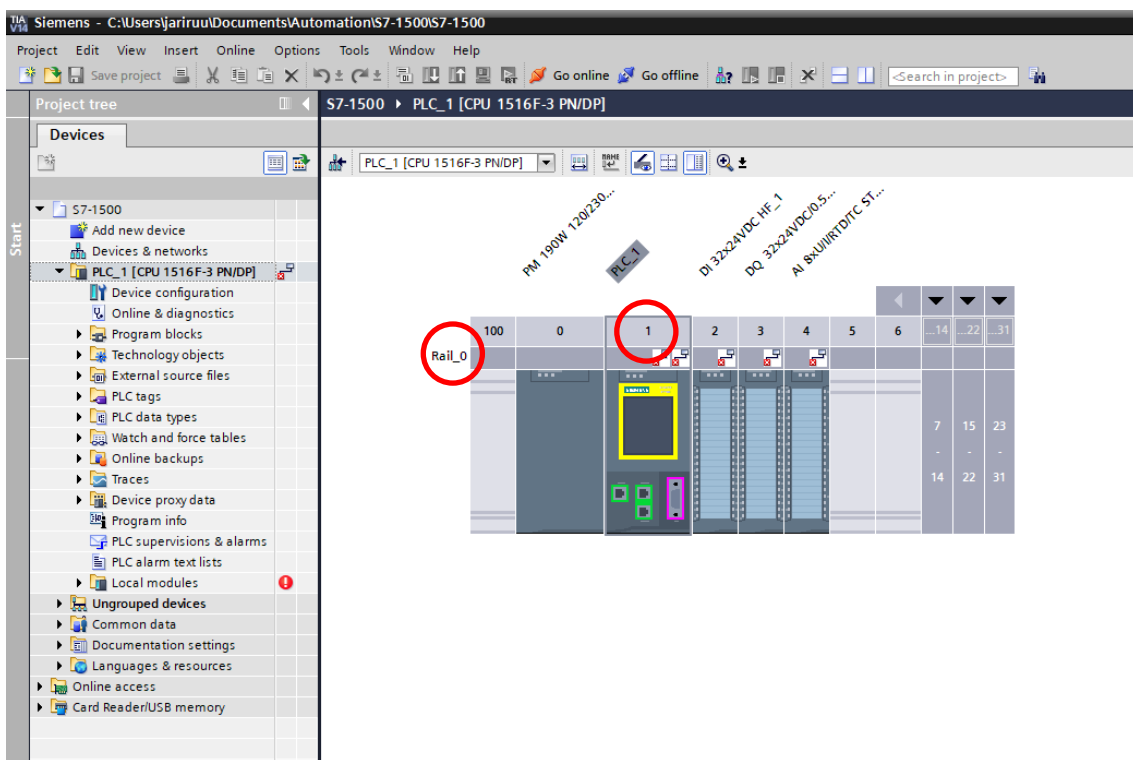
- Tämä avaa ikkunan, josta voidaan valita seuraavat vaihtoehdot (kuvat 77 ja 78):
 - o Protocol: Esimerkissä käytetään S7-protokollaa, koska logiikkana S7-1500-sarjan logiikka
 - o Name: Tähän datan lähteen nimi. Esimerkissä käytetty PLC_1.
 - o Description: Tähän voidaan kirjoittaa datan lähteen tarkemmat tiedot.
 - o Reading cycle: Oletuksena tässä on 1 m, eli dataa luetaan yhden minuutin välein datan lähteestä. Esimerkissä käytetty arvona 1 s eli dataa luetaan yhden sekunnin välein.
 - o IP address: Tähän kirjoitetaan datan lähteen IP-osoite. Esimerkkitapauksessa S7-1500-sarjan PLC-logiikan IP-osoite eli 192.168.1.1.
 - o Connection Type: Vaihtoehtoina ovat Automatic ja Manual. Valitsemalla Automatic MindSphere etsii datan lähteen teline- ja paikkanumeron automaattisesti. Valitsemalla Manual käyttäjä voi kirjoittaa datan lähteen teline- ja paikkanumeron, jotka voidaan löytää TIA Portalista PLC-projektista, missä käyttäjä on määrittänyt PLC-logiikkansa paikan. Esimerkkitapauksessa valittu Manual ja kirjoitettu Rack number -kohtaan 0 ja Slot number -kohtaan 1 (kuva 79).



Kuva 77. Dataniälhteen luonti-ikkuna.

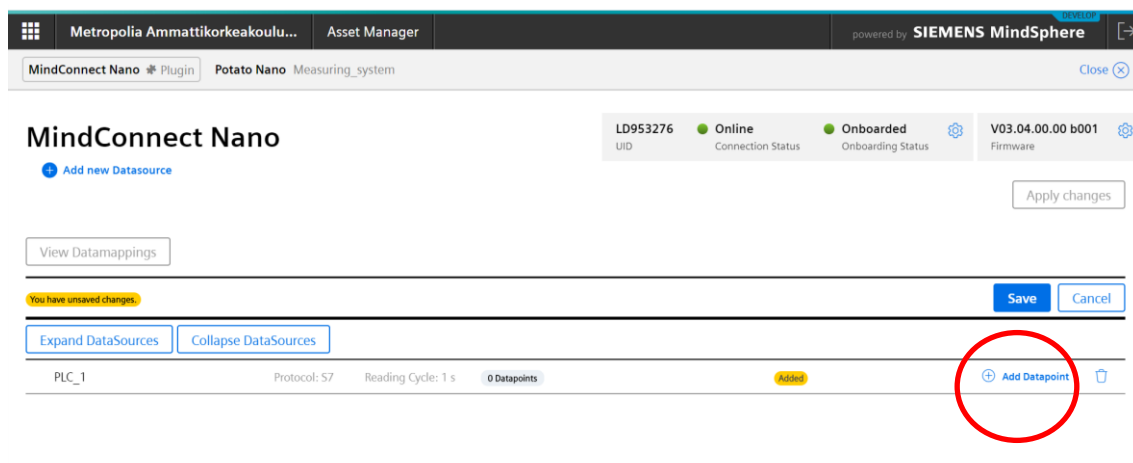


Kuva 78. Datanlähteen luonti-ikkuna.



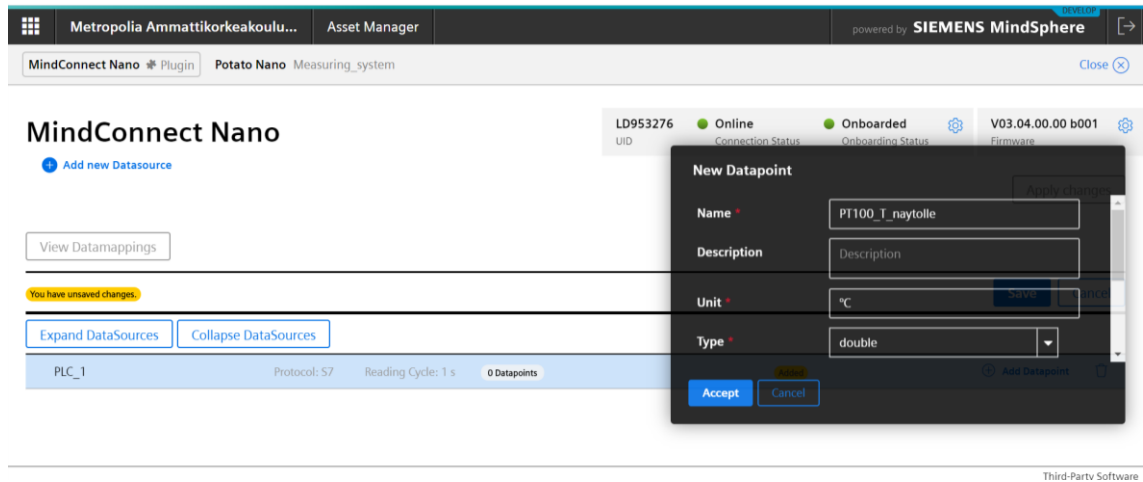
Kuva 79. PLC-projektista löytyvät Rack number ja Slot number.

- Edellä mainittujen kohtien jälkeen painetaan Accept.
- Kun käyttäjä on määrittänyt datan lähteen, painetaan siltä riviltä kohtaa Add datapoint, joka merkitty kuvassa 80 punaisella ympyrällä.

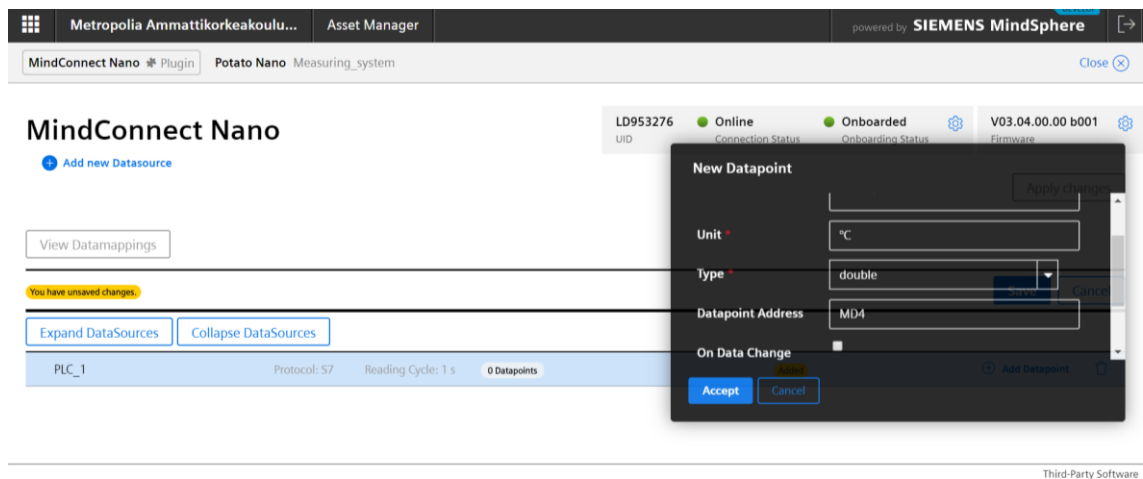


Kuva 80. Datayhteyden luonti.

- Tämä avaa ikkunan, jossa seuraavat vaihtoehdot (kuva 81 ja 82):
 - Name: Tähän määritellään data pointin nimi, joka on nimettävä samalla nimellä kuin joku niistä aiemmin aspectien luontivaiheessa tehdyistä muuttujista. Nimeäminen on tehtävä myös samalla kirjainkoolla kuin aiemmin tehdyissä muuttujissa. Esimerkissä käytetty PT100_T_naytolle-nimeämistä.
 - Description: Tähän voidaan kuvailla tarkemmin luotavaa data pointia.
 - Unit: Tähän määritellään sama Unit-tyyppi kuin aspectien luontivaiheessa. Esimerkissä muuttujalle PT100_T_naytolle määriteltiin aspectien määrittelyvaiheessa °C.
 - Type: Tähän määritellään sama type kuin aspectien luontivaiheessa. Esimerkissä aspectien määrittelyvaiheessa käytettiin muuttujalle PT100_T_naytolle muuttujalle arvoa Double.
 - Datapoint address: Tähän määritellään muuttujalle sen osoite PLC-logiikalla data pointin luomiseksi. Esimerkissä PT100_T_naytolle-muuttujalle käytettiin PLC-ohjelmassa muuttujaa MD4, joten laitetaan se tähän. Muuttujien PLC-logiikan osoitteet näkyvät kuvassa 47.
 - On Data Change: Mikäli tähän kohtaan laittaa rastin, voidaan määritellä data pointin hystereesi.

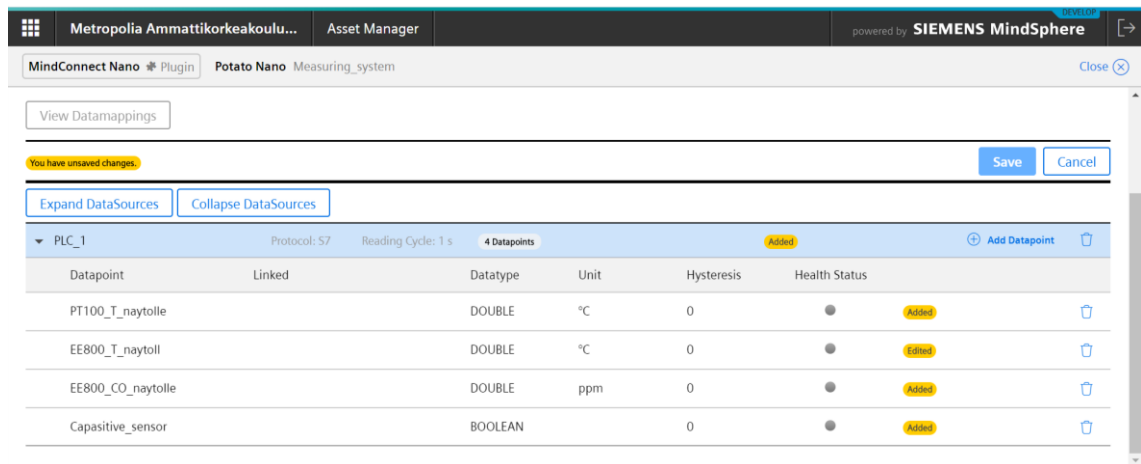


Kuva 81. Datayhteyden luonti-ikkuna.



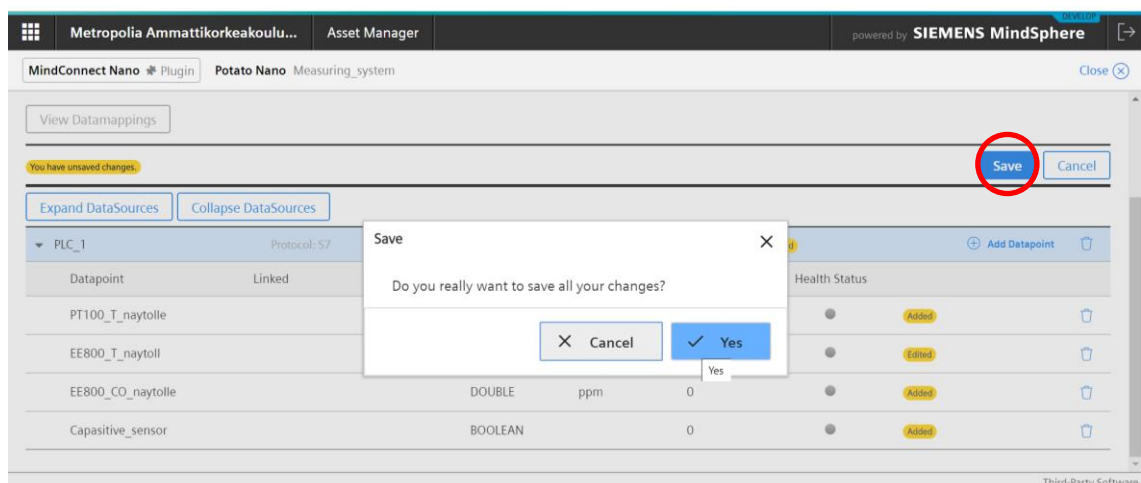
Kuva 82. Datayhteyden luonti-ikkuna.

- Datayhteyden luomiseksi painetaan lopuksi Accept.
- Esimerkissä luodaan vielä muuttujille EE800_T_naytolle, EE800_CO_naytolle ja Capacitive_sensor omat data pointit. Datapoint address -osoitteena on käytetty niissä kuvan 47 mukaisesti MD12, MD20 ja IO.0 (kuva 83).



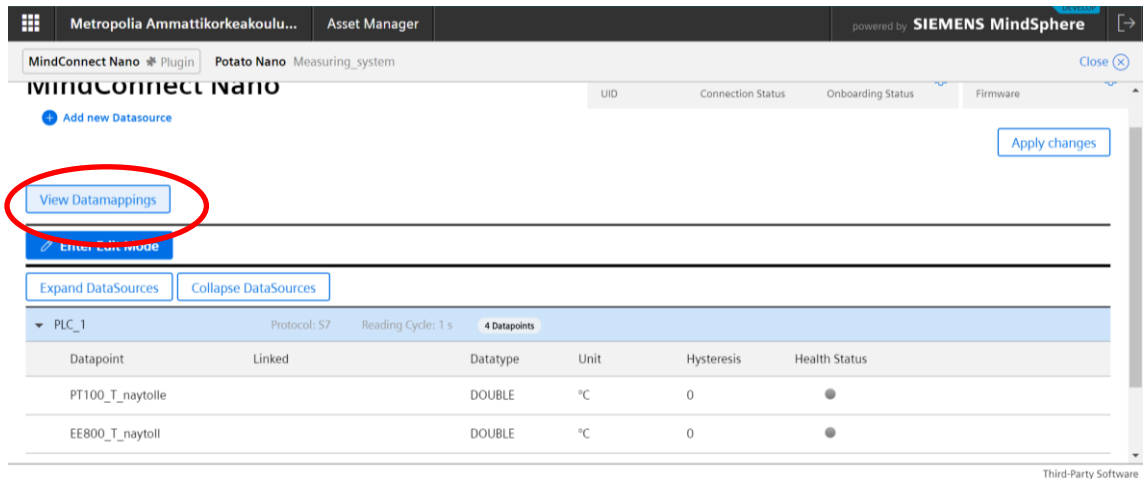
Kuva 83. MindConnect Nano-yhteysikkuna esimerkkinuuttujen lisäämisen jälkeen.

- Lopuksi tallennetaan luodut konfiguraatiot MindConnect Nano -yhteysikkunassa Save-näppäimestä, joka tuplavarmistaa vielä valinnan (kuva 84). Punaisella ympyrällä merkitty tallennusnäppäin.



Kuva 84. Datanlähteen ja datayhteyksien jälkeinen tallennus.

- Tämän jälkeen painetaan View Datamappings-painiketta, joka mahdollistaa data pointien ja aspectien muuttujien välisen linkkauksien. Näppäin merkitty kuvassa 85 punaisella ellipsillä.



Kuva 85. Datankartoituksen valinta.

- Painetaan muuttujan Link Variable -painiketta. Esimerkissä painetaan muuttujan PT100_T_naytolle Link Variable -painiketta (kuva 86).
 - Tämän jälkeen etsitään avautuvasta ikkunasta data pointia vastaava muuttuja (kuva 87) ja painetaan muuttujan vieressä olevaa Link variable -painiketta, jolloin kohtaan ilmestyy pallo. Lopuksi painetaan Accept. Esimerkissä etsitään PT100_T_naytolle-data pointin vastaava muuttuja.
 - Esimerkitapauksessa etsitään vielä lopuille data pointeille omat muuttujat, jonka jälkeen Datamappings-ikkuna näyttää, että datapointin ja muuttujat ovat linkattu toisiinsa (kuva 88).
 - Mikäli jostain syystä data pointia ei pysty linkkaamaan muuttujaan, kannattaa tarkastaa data pointin kirjoitusmuoto, niiden pitää vastata muuttujan vastaavia tietoja. Esimerkiksi isot kirjaimet vaikuttavat muuttujan löytymiseen.

Datapoint	Mapping
PT100_T_naytolle °C, DOUBLE	
EE800_T_naytoll °C, DOUBLE	
EE800_CO_naytolle ppm, DOUBLE	
Capacitive_sensor , BOOLEAN	

Kuva 86. Datankartoitus-ikkuna.

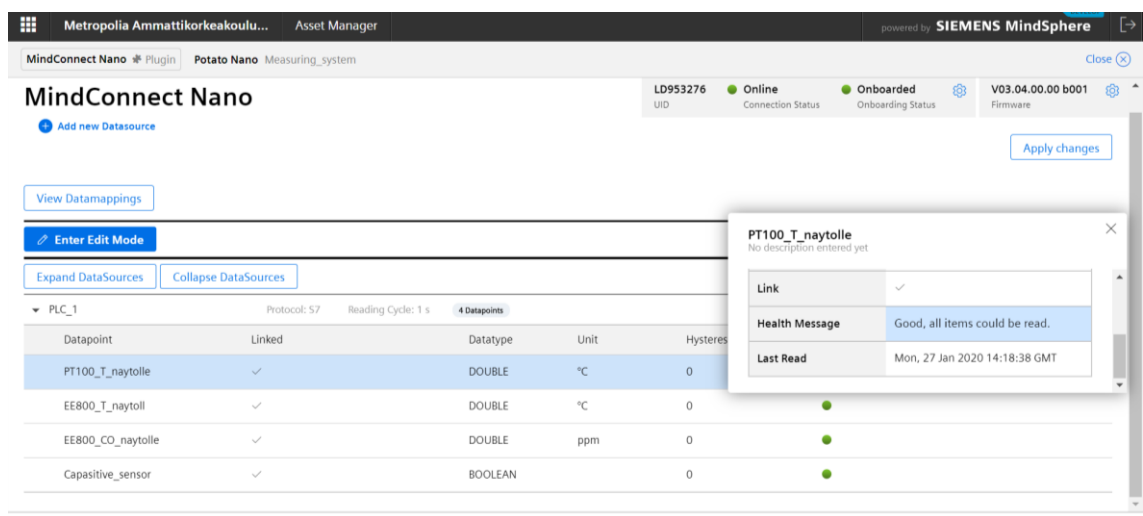
Name	Type	Unit
EE800_T_naytolle	DOUBLE	°C
PT100_T_naytolle	DOUBLE	°C
firmwareStatus	0 of 3 compatible	0 maps for 3 links

Kuva 87. Datanlinkkaus-ikkuna.

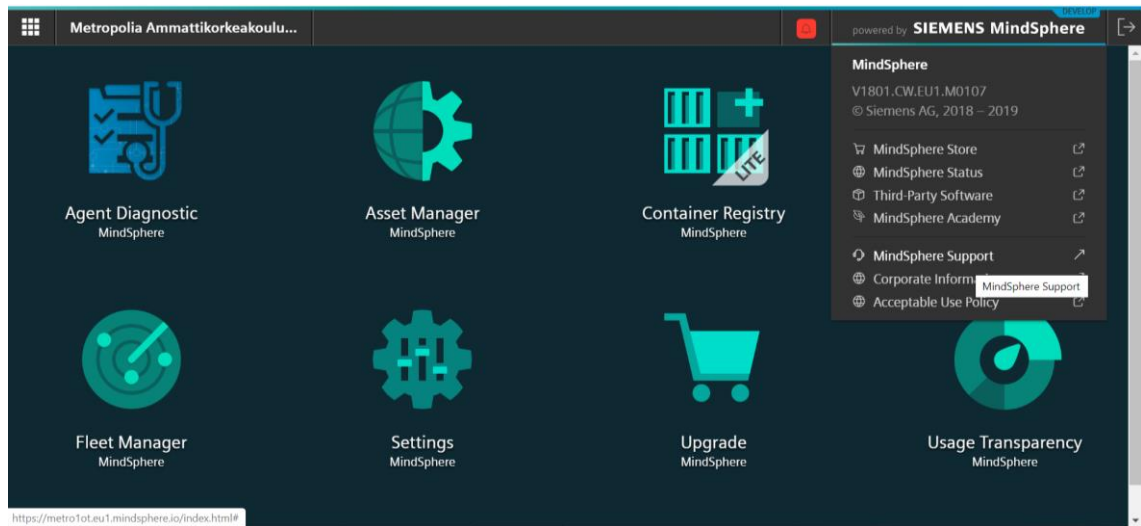
Datapoint	Mapping
PT100_T_naytolle °C, DOUBLE	Potato Nano > Temperature > PT100_T_naytolle
EE800_T_naytoll °C, DOUBLE	Potato Nano > Temperature > EE800_T_naytolle
EE800_CO_naytolle ppm, DOUBLE	Potato Nano > Carbon_dioxide > EE800_CO_naytolle
Capacitive_sensor , BOOLEAN	Potato Nano > Proximity_IOs > Capacitive_sensor

Kuva 88. Datankartoitus-ikkuna datanlinkkauksien jälkeen.

- Mikäli kaikki on tehty oikein, MindConnect Nano -yhteysikkunan pitäisi näyttää kuvan 89 kaltaiselta, jossa Data pointien Health status -valot pitäisivät olla vihreitä. MindSpheren ja MindConnectin välinen yhteydenotto kestää hetken, joten kannattaa päivitellä selainta, mikäli pallot eivät muutukaan heti vihreiksi. Palloja painamalla ilmestyy viesti, jossa lukee Health message -kentässä Good, all items could be read.
 - o Mikäli pallot ovat punaisia, yhteydessä tai konfiguraatiossa on joku ongelma. Tällöin kannattaa tarkastella Getting Connected to MindSphere -ohjetta, jos ongelma ei tätä ohjetta tarkastellen ratkea. Viime kädessä Siemensin oma tukipalvelu löytyy kuvan 90 mukaisesti.

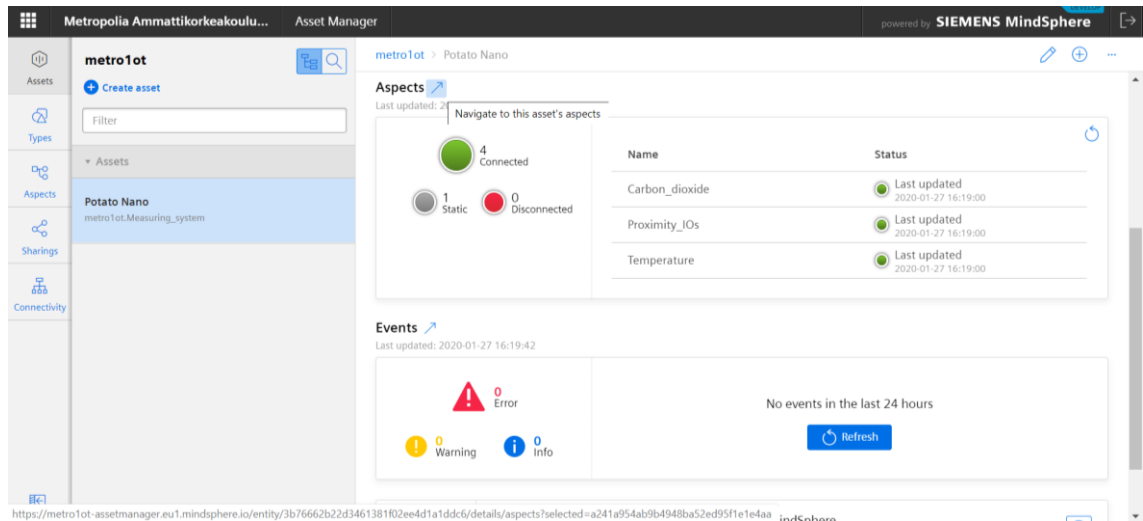


Kuva 89. MindConnect Nano -yhteysikkuna, datalinkkauksien jälkeen.

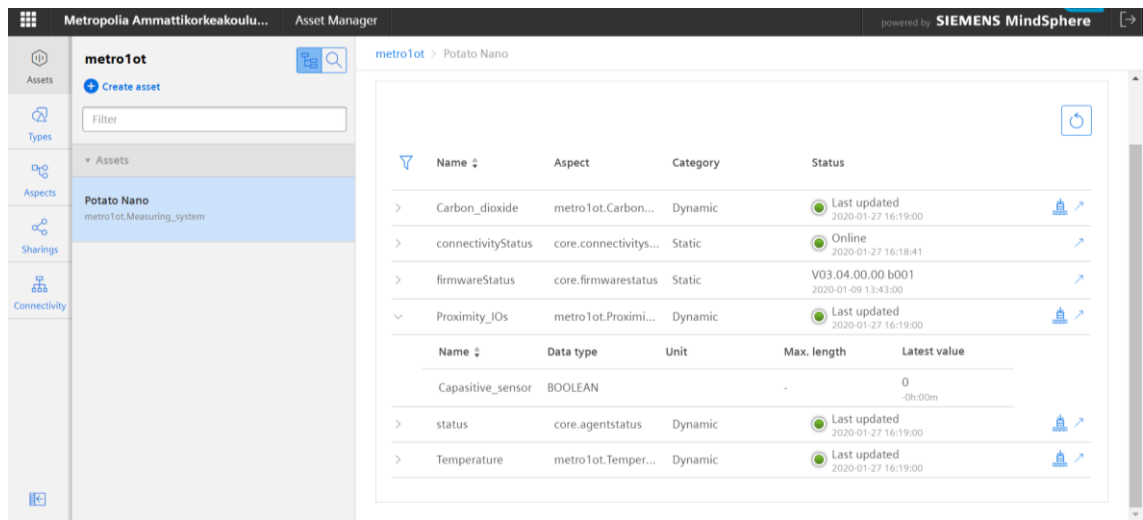


Kuva 90. MindSpheren aloitussivu, josta löytyy ylävalikosta yhteys MindSpheren tukipalveluun.

- Mikäli halutaan tarkastella, että mitä hetkellisiä arvoja MindSphere vastaanottaa PLC-logiikan muuttujilta, mennään oman luodun assetin kohtaan Aspects ja painetaan sen edessä olevaa nuolta (kuva 91).
 - o Aspects-ikkuna avautuu, joka näyttää tämän assetin aspectit. Esimerkissä Proximity_IOs-aspect avataan rivillä olevasta nuolesta, joka näyttää aspectin muuttujan, sen tyyppin ja viimeisimmän arvon. Capacitive_sensor-muuttujan arvo on 0 (kuva 92).
 - o Muuttujan arvo muuttuu sen tilan vaihtuessa PLC-logiikalla. Tilan muutos näkyy hetken kuluttua myös MindSpheressä (kuva 93).



Kuva 91. Asset-ikkuna yhteyksien luonnin jälkeen.



Kuva 92. Assetin aspectit yhteyksien luonnin jälkeen.

Name	Aspect	Category	Status	
Carbon_dioxide	metro1ot.Carbon...	Dynamic	Last updated 2020-01-27 16:21:00	
connectivityStatus	core.connectivitys...	Static	Online 2020-01-27 16:18:41	
firmwareStatus	core.firmwarestatus	Static	V03.04.00.00 b001 2020-01-09 13:43:00	
Proximity_IOs	metro1ot.Proximi...	Dynamic	Last updated 2020-01-27 16:21:00	
Name	Data type	Unit	Max. length	Latest value
Capasitive_sensor	BOOLEAN		-	1 -0h:00m
status	core.agentstatus	Dynamic		Last updated 2020-01-27 16:21:00
Temperature	metro1ot.Temper...	Dynamic		Last updated 2020-01-27 16:21:00

Kuva 93. Capasitive_sensor- muuttujan arvo on muuttunut kuvaan 92 verrattuna.

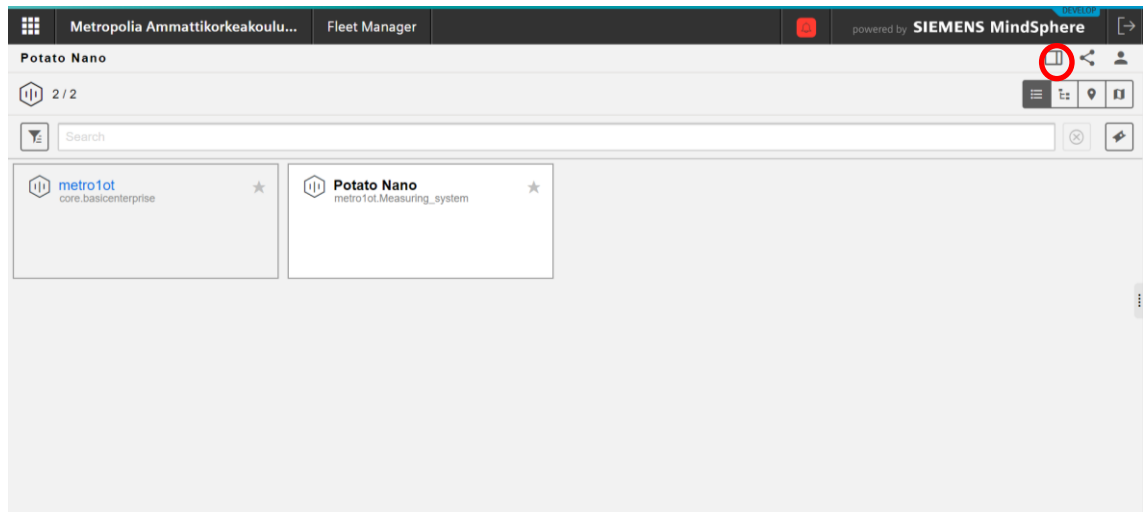
- Kuvassa 94 näkyy kaikki esimerkin aspectien muuttujat toiminnassa.

Name	Aspect	Category	Status	
Carbon_dioxide	metro1ot.Carbon_dioxide	Dynamic	Last updated 2020-01-29 15:29:00	
Name	Data type	Unit	Max. length	Latest value
EE800_CO_naytoille	DOUBLE	ppm	-	456.95892 -0h:00m
connectivityStatus	core.connectivitystatus	Static		Online 2020-01-29 15:28:18
firmwareStatus	core.firmwarestatus	Static		V03.04.00.00 b001 2020-01-09 13:43:00
Proximity_IOs	metro1ot.Proximity_IOs	Dynamic	Last updated 2020-01-29 15:29:00	
Name	Data type	Unit	Max. length	Latest value
Capasitive_sensor	BOOLEAN		-	1 -0h:00m
status	core.agentstatus	Dynamic		Last updated 2020-01-29 15:29:00
Temperature	metro1ot.Temperature	Dynamic		Last updated 2020-01-29 15:29:00
Name	Data type	Unit	Max. length	Latest value
EE800_T_naytoille	DOUBLE	°C	-	22.437428 -0h:00m
PT100_T_naytoille	DOUBLE	°C	-	21.507162 -0h:00m

Kuva 94. Aspectien muuttujat näyttävät arvoja.

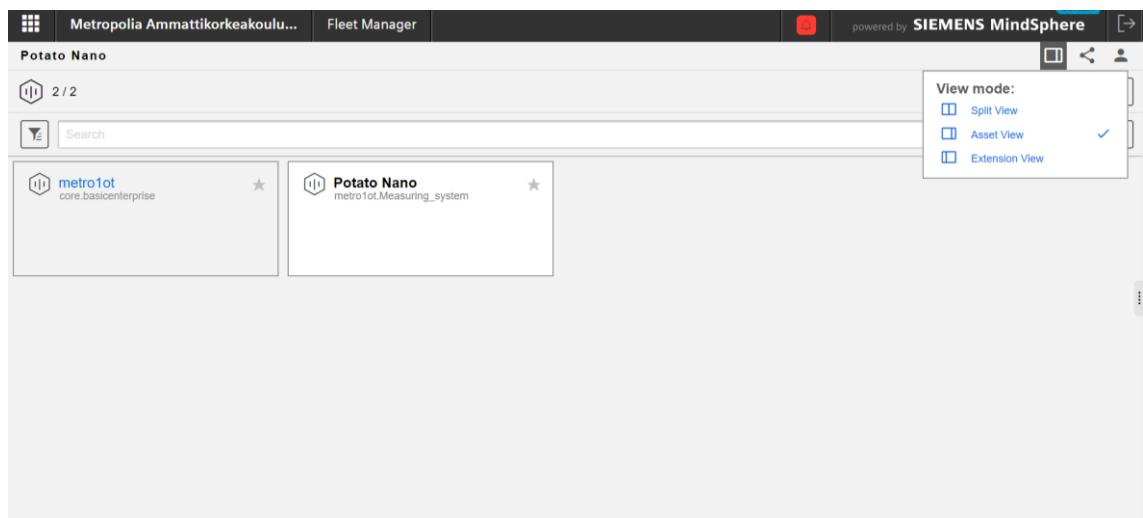
3.6 Fleet Managerin käyttöönotto

- Valitaan MindSpheren aloitusruudulta Fleet Manager -sovellus.
- Avautuvasta kuvan 95 kaltaisesta näkymästä valitaan punaisella ympyrällä merkitty painike.



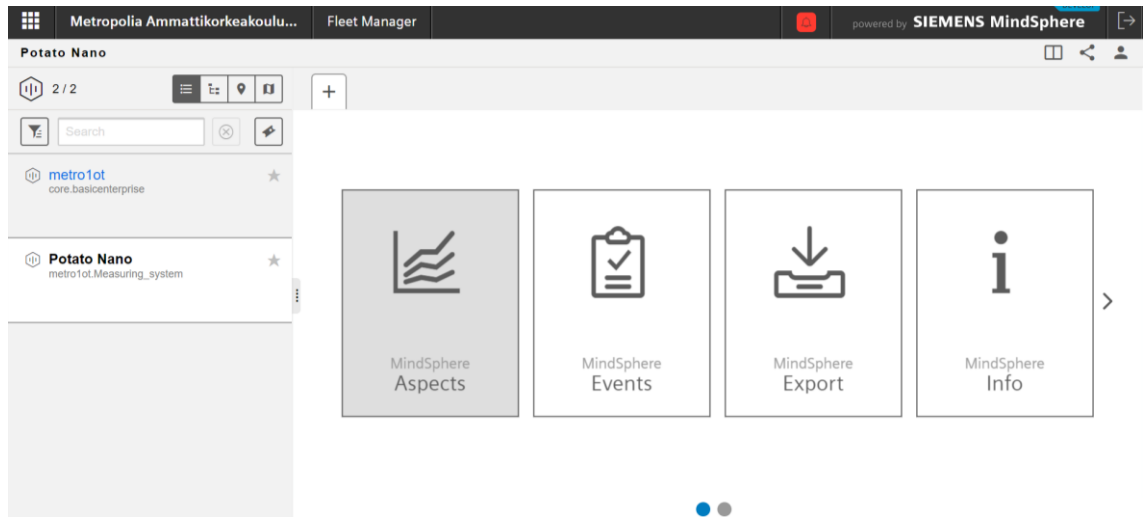
Kuva 95. Fleet Manager -ikkuna.

- Avautuvasta valinta-ikkunasta valitaan Split View (kuva 96).



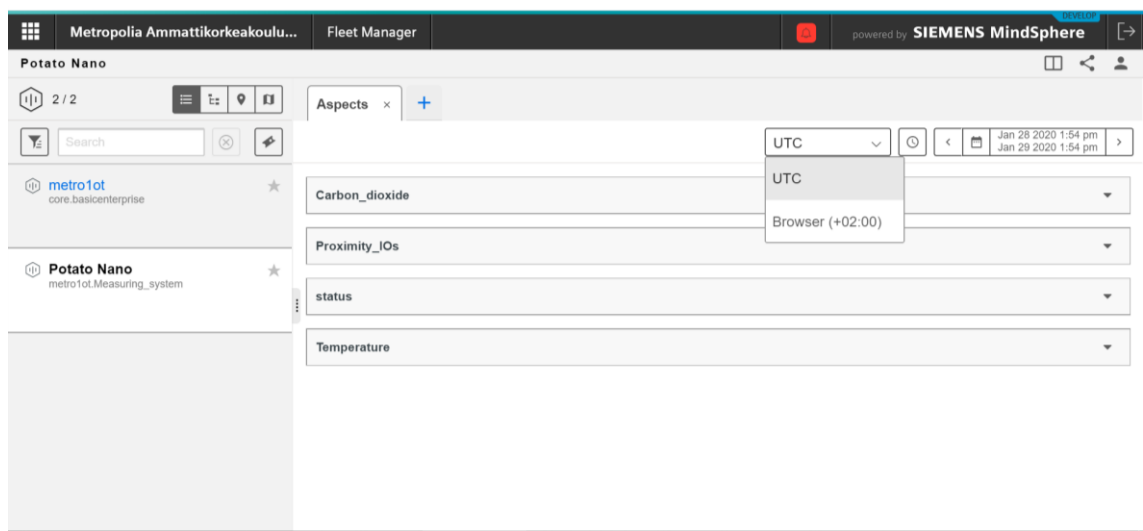
Kuva 96. Katselumoodin valinta.

- Valitaan vasemmalta haluttu asset tummaksi, esimerkkitapauksessa Potato Nano. Tämän jälkeen valitaan keskeltä MindSphere Aspects -painike (kuva 97).



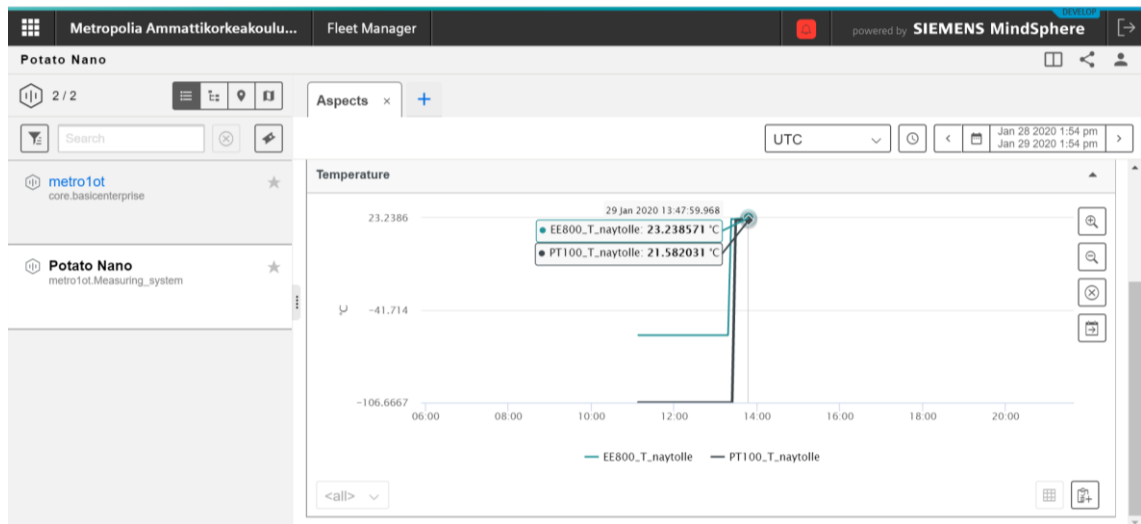
Kuva 97. Aspecttien historiokuvaajien valinta.

- Avautuva näkymä (kuva 98) näyttää valitun assetin dynaamiset aspectit. Aspecttien yläpuolelta voidaan valita, että mitä aikavyöhykettä käytetään sekä miltä ajanjaksolta historiatietoja näytetään.



Kuva 98. Valitun assetin dynaamiset aspectit.

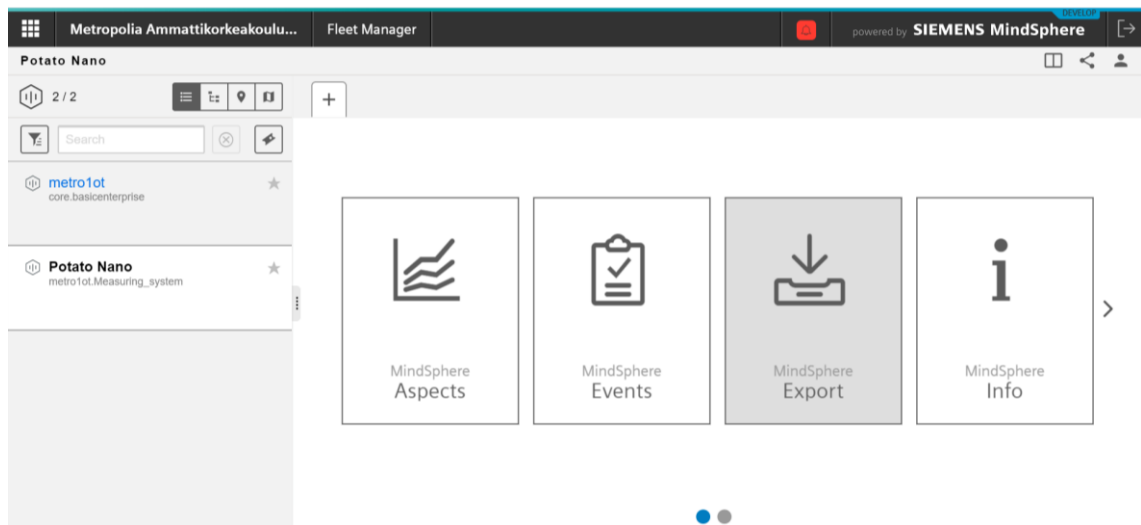
- Esimerkitapauksen Temperature-aspectin muuttujien historiakäyrät lyhyeltä ajanjaksolta (kuva 99).



Kuva 99. Temperature-aspectin muuttujien historiokuvaajat.

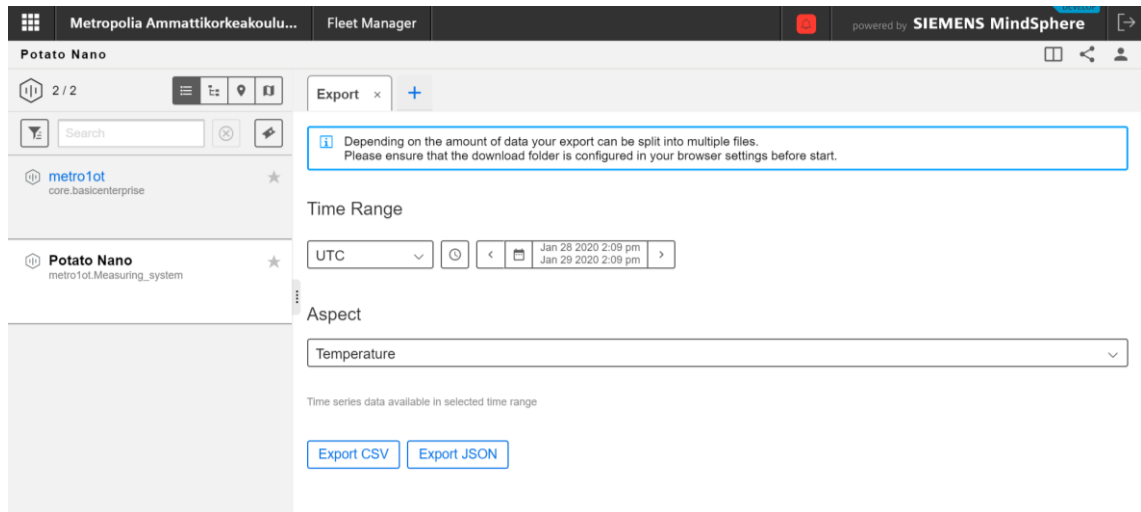
3.6.1 Datan export-toiminto Fleet Managerissa

- Kuvan 100 mukaisesta näkymästä assetin ollessa valittuna ja painettaessa Export-painiketta päästään Export-valikkoon.



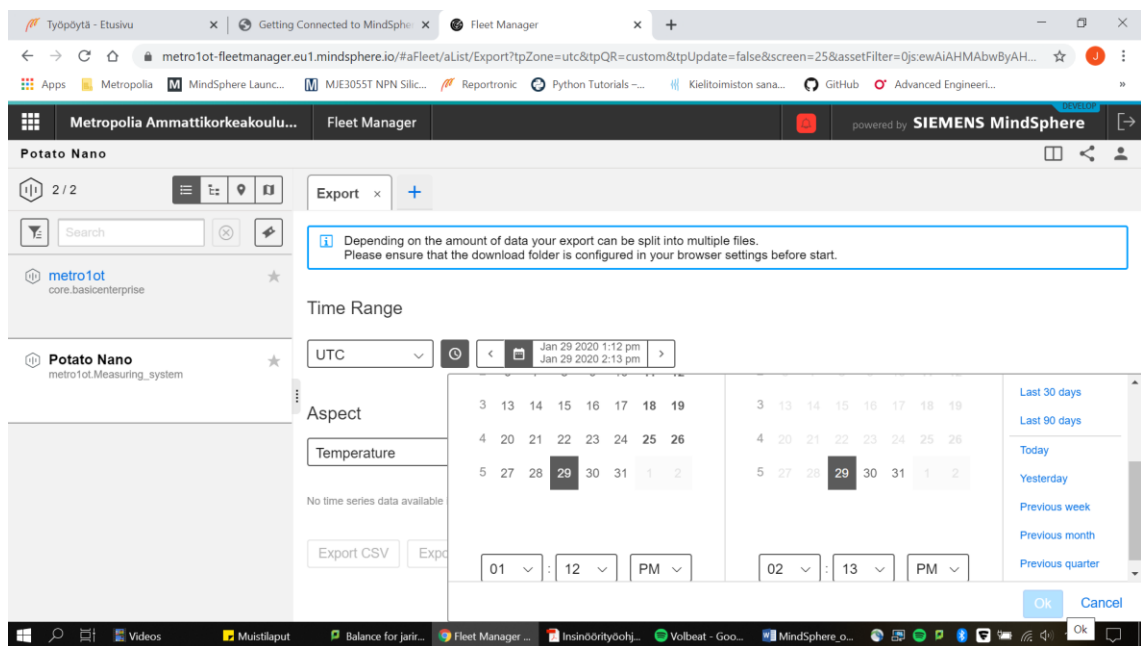
Kuva 100. Export-valikon valinta.

- Kuvan 101 mukaisesta näkymästä voidaan valita datan export-ajanjakso sekä haluttu aspect mistä data exportataan.



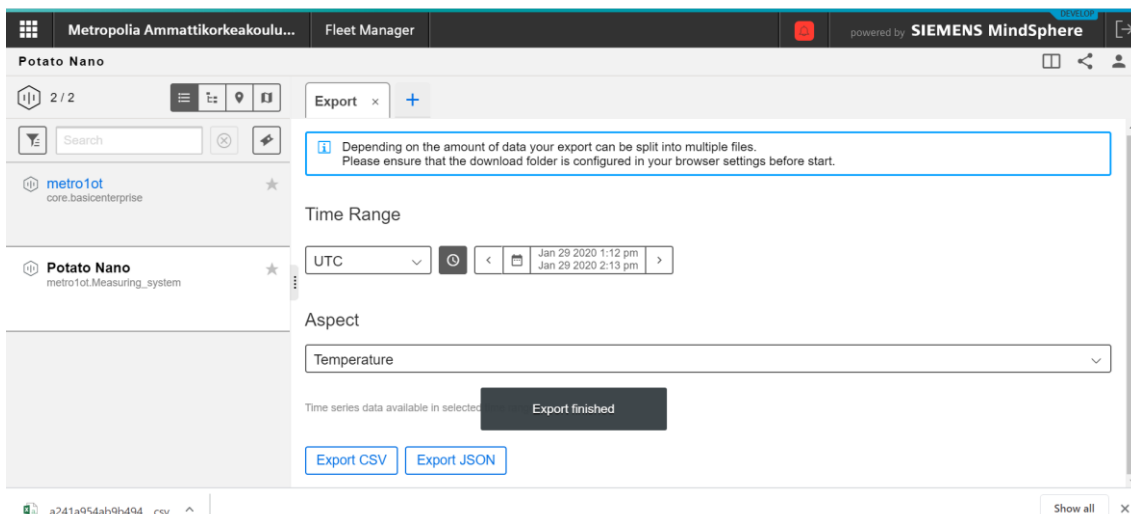
Kuva 101. Export-näkymä.

- Esimerkissä valitaan datan export-ajanjaksoksi tunnin pituinen aika (kuva 102).



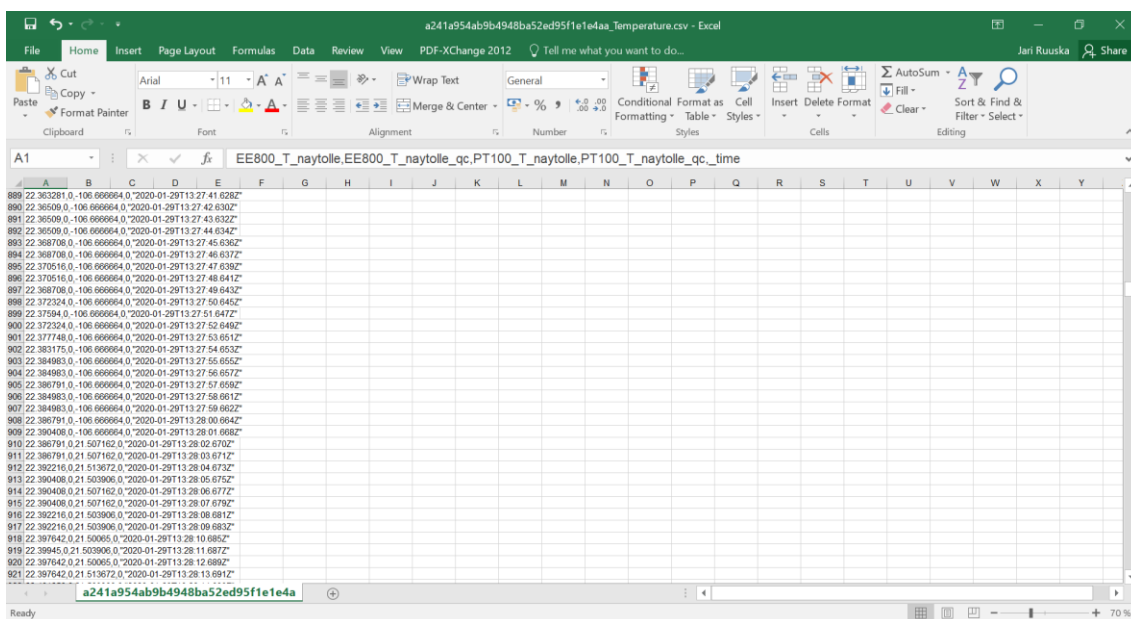
Kuva 102. Datan export-ajanjakson valinta.

- Tämän jälkeen painetaan haluttua export-painiketta. Esimerkissä valitaan Microsoft Excel -tiedoston tuovaa Export CSV -painiketta. Tämä valinta lataa suoraan tiedoston tietokoneelle (kuva 103).



Kuva 103. Export-tapahtuma.

- Kuvassa 104 nähdään avattu Excel-tiedosto, jossa dataa joudutaan vielä muokkaamaan haluttuun muotoon.



Kuva 104. Exportattu Excel-tiedosto.

4 Loppusanat

Tässä ohjeessa ja sitä käsitelleessä projektissa otettiin yhteys MindSpheren, MindConnect-laitteen ja projektissa käytetyn PLC-logiikan välille. Ohje kirjoitettiin onnistuneen projektin vaiheiden mukaisesti.

Lähteet

- 1 What is the Internet of Things (IoT)? 2019. Verkkoaineisto. Red Hat. <<https://www.redhat.com/en/topics/internet-of-things/what-is-iot>>. 11.9.2019. Luettu 31.1.2020.
- 2 Calum McClelland. 2019. How Does an IoT Solution Actually Work? Verkkoaineisto. IoT for all. <<https://www.iotforall.com/iot-explained-how-does-an-iot-system-actually-work/>>. 6.1.2019. Luettu 6.2.2020.
- 3 S7 Communication (S7comm). 2016 Verkkoaineisto. Wireshark. <<https://wiki.wireshark.org/S7comm>>. 13.5.2016. Luettu 8.2.2020.
- 4 Sean Doyle. 2018. TCP vs. UDP: Understanding the Difference. Verkkoaineisto. Privacy News Online. <<https://www.privateinternetaccess.com/blog/2018/12/tcp-vs-udp-understanding-the-difference/>>. 17.12.2018. Luettu 6.2.2020.
- 5 Digitalisoi liiketoimintasi MindSphere IoT-alustalla. 2017. Verkkoaineisto. Siemens. <<https://new.siemens.com/fi/fi/tuotteet/teollisuus/mindsphere.html>>. 5.9.2017. Luettu 28.1.2020.
- 6 Getting Connected to MindSphere. 2020. PDF-dokumentti. Siemens. 7.1.2020. Luettu 28.01.2020.
- 7 Introduction to netstat. 2019. Verkkoaineisto. IONOS. <<https://www.ionos.com/digitalguide/server/tools/introduction-to-netstat>>. 6.3.2019. Luettu 29.1.2020.
- 8 MindConnect Nano — Firmware and Open Source Software (OSS). 2020. Verkkoaineisto. Siemens. <[https://support.industry.siemens.com/cs/document/109745561/mindconnect-nano-%E2%80%93-firmware-and-open-source-software-\(oss\)?dti=0&lc=en-WW](https://support.industry.siemens.com/cs/document/109745561/mindconnect-nano-%E2%80%93-firmware-and-open-source-software-(oss)?dti=0&lc=en-WW)>. 14.1.2020. Luettu 29.1.2020.
- 9 MindConnect IoT2040 — Firmware and Open Source Software (OSS). 2020. Verkkoaineisto. Siemens. <[https://support.industry.siemens.com/cs/document/109745562/mindconnect-iot2040-%E2%80%93-firmware-and-open-source-software-\(oss\)?dti=0&lc=en-WW](https://support.industry.siemens.com/cs/document/109745562/mindconnect-iot2040-%E2%80%93-firmware-and-open-source-software-(oss)?dti=0&lc=en-WW)>. 14.1.2020. Luettu 29.1.2020