



Vaasan yliopisto  
UNIVERSITY OF VAASA

Aapo Kirsilä

# **Avoimen lähdekoodin ratkaisun kaupallistaminen palvelumallin avulla**

Case yritys

Tekniikan ja innovaatiojohtamisen  
akateeminen yksikkö  
Pro-Gradu -tutkielma  
Tietojärjestelmätiede

Vaasa 2025

---

**VAASAN YLIOPISTO****Tekniikan ja innovaatiojohtamisen akateeminen yksikkö**

<b>Tekijä:</b>	Aapo Kirsilä		
<b>Tutkielman nimi:</b>	Avoimen lähdekoodin ratkaisun kaupallistaminen palvelumallin avulla : Case yritys		
<b>Tutkinto:</b>	Kauppätieteiden maisteri		
<b>Oppiaine:</b>	Tietojärjestelmätiede		
<b>Työn ohjaaja:</b>	Tero Vartiainen		
<b>Valmistumisvuosi:</b>	2025	<b>Sivumäärä:</b>	33

---

**TIIVISTELMÄ:**

Tämän työn tarkoituksena on esitellä avoimen lähdekoodin ohjelmiston ja teknologian kaupallistamisen haasteita ja mahdollisuuksia. Työssä on tapaustutkimuksen tapauskohteena Ardu-Pilot-ohjelmisto alustaa ja Pixhawk-robotialustaa käyttävä yritys x. Lisäksi sen kolme erilaista tuotetta, joilla kaikilla on erilaiset tarpeet ja liiketoimintamallit. Perimmäinen ongelma avoimen lähdekoodin käytössä on koodin avoimuus, jolloin kuka tahansa voi alkaa käyttää vastaavaa tekniikkaa ja myydä vastaavia tuotteita omalla brändillään. Samaan aikaan avoimuus on myös vahvuus ja mahdollistaa erittäin nopean ja ketterän kehityksen. Erilaiset avoimen lähdekoodin lisenssit tarjoavat erilaiset mahdollisuudet kehittää ja kaupallistaa alustaa. Lisäksi erilaiset palvelumallit mahdollistavat tuotteen kaupallistamisen riippuen siitä, minkälainen tuote tai palvelu on kyseessä.

Tapaustutkimuksessa tarkastellaan kolmea eri liiketoimintamallia, joita verrataan olemassa oleviin, vertailukelpoisiin ratkaisuihin markkinoilla. Yrityksen tarjoamien palveluiden lähtökohdaksi on tuottaa asiakkaalle mahdollisimman suurta arvoa. Teknologisesti kyseessä on kyberfysinen järjestelmä, jossa erilaiset sensorisyötteet muunnetaan ohjauksikäskyiksi fyysisille komponenteille. Tuotteen perustana toimii sekä tuotemalli että palvelumalli, joita voidaan yhdistellä eri tavoin tuotekohtaisesti.

Avoimen lähdekoodin lisenssit muodostavat erilaisia ongelmia kaupallistamisen tielle, koska tietyt lisenssit rajoittavat lähdekoodin käyttöä kaupallisesti ja toiset taas vaativat pitämään sen avoimena. Lisäksi samat ongelmat seuraavat myös avoimien laitteistojen mukana. Tässä työssä on esitelty käytössä olevien ohjelmistojen ja laitteistojen lisenssit ja miten niitä voi käyttää kaupallisessa käytössä. Lisäksi tässä työssä on esitelty käytössä olevien ohjelmistojen vaatimukset niiden eri käyttötarkoituksissa.

Lean startup metodin ja avoimen lähdekoodin avulla voidaan tuotteita saada markkinoille hyvinkin nopeasti ja voidaan löytää uusia käyttötapauksia yhdessä asiakkaiden kanssa. Lisäksi sen avulla voidaan aloittaa jatkuvan kehityksen tie, jossa asiakkaiden vaatimukset pyritään jatkuvasti täyttämään. Avoin lähdekoodi mahdollistaa nopean ja ketterän tavan kehittää uusia tuotteita, koska koko kehitystä ei tarvitse aloittaa alusta. Avoimen lähdekoodin yhteisön tuki voi olla alkuvaiheen ongelmissa elintärkeää. Samalla voi kuitenkin olla vaikea hahmottaa miksi joku asia tapahtuu tai toimii tietyllä tavalla, koska kehitys on hajautettu ja kehittäjät kehittävät alustaa aina omalla tavallaan.

---

**AVAINSANAT:** Avoin lähdekoodi, palvelullistaminen, liiketoimintamallit, lisenssit

## Sisällys

1	Johdanto	5
1.1	Tutkimuksen tavoite ja rajaus	5
1.2	Tutkimusmenetelmä	5
1.3	Tutkimuskysymys	8
2	Teoreettinen viitekehys	9
2.1	Lean startup metodi	9
2.2	Avoin lähdekoodi	9
2.3	Avoimet laitteistot	11
2.4	Palvelumallit asiakkaan ja palveluntarjoajan näkökulmasta	12
2.4.1	Tuotemalli, jossa tuotteilla on rajoitettu tuki	13
2.4.2	Asennetut ja tuetut tuotteet	13
2.4.3	Täydentävät palvelut	14
2.4.4	Tuotteeseen keskittyvät palvelut	14
2.4.5	Leasing	15
2.4.6	Käyttöpalvelu	15
2.4.7	Hallittu palveluratkaisu	15
2.4.8	Kokonaisratkaisu	16
3	Tapaustutkimuskohteen tapauksen ratkaisu	17
3.1	TUOTE 1 Automaattiohjaus jyrään	17
3.2	TUOTE 2 Kuusikopteri automaattiajolla	19
3.3	TUOTE 3 Autonominen saunalautta	21
3.3.1	Turvallisuus	22
3.3.2	3D-kamera OAK depthAi	22
3.3.3	Kilpaileva alusta Ros2	23
4	Tulokset	24
5	Diskussio	28
5.1	Johtopäätökset	28
	Lähteet	30

## **Kuvat**

Kuva 1 Tuote 1 Autonominen jyrä	18
Kuva 2 Tuote 2 Autonominen kuusikopteri	19
Kuva 3 Käyttöliittymä taulutietokoneella	20
Kuva 4 Tuote 3 Autonominen saunalautta	21

## **Taulukot**

Taulukko 1 Tutkimuskysymys	8
Taulukko 2 Tulokset tuotteelle 1	25
Taulukko 3 Tulokset tuotteelle 2	26
Taulukko 4 Tulokset tuotteelle 3	27

## **Lyhenteet**

**MVP (Minium valuable product)** pienin julkaistavissa oleva tuote  
**Ros2 (The Robot Operating System)** Robottikäyttöjärjestelmä  
**Lidar (light detection and ranging)** valotutka  
**PCS (Cyber-physical systems)** Kyberfyysiset järjestelmät

# 1 Johdanto

## 1.1 Tutkimuksen tavoite ja rajaus

Tämän työn tarkoituksena on esitellä avoimen lähdekoodin ohjelmiston ja teknologian kaupallistamisen haasteita ja mahdollisuuksia. Työssä käytetään tapaustutkimuksen kohteena Ardupilot-ohjelmistoa ja Pixhawk-robotialustaa käyttävää yritystä, josta käytetään nimeä yritys x. Yritys toimii Lean startup menetelmän mukaisella metodilla, jossa keskiössä ovat avoin lähdekoodi ja avoimet laitteistot. Perimmäinen ongelma avoimen lähdekoodin käytössä on koodin avoimuus, jolloin kuka tahansa voi alkaa käyttää vastaavaa tekniikkaa ja myydä vastaavia tuotteita omalla brändillään. Monissa tapauksissa tehdään sivuhaara ja suljetaan oma koodi avoimen koodin ulkopuolelle. Tässä haasteeksi tulee jatkokehitys, koska jokaisen päivityksen jälkeen sivuhaara on saatava toimimaan avoimen koodin kanssa yhteen. Yksi ratkaisu on tarjota alusta palveluna niin houkuttelevaan hintaan, ettei kilpailijan kannata nähdä vaivaa tarjotakseen vastaavaa tuotetta. Lisäksi tässä työssä esitellään olemassa olevia kaupallisia ratkaisuja, joiden pohjana on avoin lähdekoodi. Palvelumalleihin ja niiden etuihin pureudutaan syvemmin. Erilaisia avoimen lähdekoodin lisenssejä ja niiden etuja avataan työssä yrityksen näkökulmasta.

## 1.2 Tutkimusmenetelmä

Tapaustutkimusmenetelmä on laadullinen tutkimusmenetelmä, joka keskittyy syvälliseen ja kattavaan tutkimukseen yhdestä tai useammasta tapauksesta. Tässä menetelmässä tutkitaan perusteellisesti tiettyjä yksilöllisiä tapauksia, ilmiöitä, organisaatioita tai prosesseja. Tapaustutkimuksessa pyritään ymmärtämään tapausten monimuotoisuutta ja kontekstuaalisia tekijöitä sekä luomaan syvällistä tietoa tutkittavasta ilmiöstä.

Tapaustutkimuksen tarkka määrittely on hankalaa, koska sitä voidaan toteuttaa monin eri tavoin ja monilla eri tieteenaloilla. Oleellista kuitenkin on, että kokonaisuus muodostuu eri aineistoista ja lopuksi muodostaa yhtenäisen kokonaisuuden eli tapauksen. Tyypillisesti tapaustutkimus muodostuu yksittäisistä tapauksista tai pienestä keskenään vuorovaikutuksessa olevasta tapausten joukosta, josta pyritään tuottamaan yksityiskohtaista ja syvällistä tietoa. Menetelmän heikkoutena on pidetty sen edustavuuden puutteesta sekä aineiston analyysin ja keruun heikosta systemaattisuudesta. Kuitenkin menetelmän vahvuutena pidetään sen kokonaisvaltaisuutta (Valli & Aarnos, 2018, s.190–191).

Tapaustutkimus voidaan ymmärtää tutkimustapana tai -strategiana, jossa hyödynnetään erilaisia menetelmiä ja aineistoja. Tarkasteltavana kohteena voi olla yksittäinen tapaus, ilmiö tai tapahtumakulku. Siinä siis voidaan tarkastella tapausjoukkoa tai vaan yhtä tapausta (Laine, Bamberg & Jokinen 2007, s.9).

Tapaustutkimuksessa tutkitaan vain yhtä tai useampaa tapausta. Tutkimuksen keskeisenä tavoitteena on tapauksen määrittely, analysointi ja ratkaiseminen. Tyypillistä tapaustutkimukselle on laadullisen aineiston käyttö, mutta myös määrällistä aineistoa voidaan hyödyntää. Aineiston analysointiin voidaan soveltaa erilaisia menetelmiä. Tapaustutkimus soveltuu erityisesti tilanteisiin, joissa tarkastellaan ajankohtaista ilmiötä, tutkijalla on rajallinen mahdollisuus kontrolloida tapahtumia, aiheesta on vähän aiempaa tutkimusta tai kun tutkimuskysymykset liittyvät siihen, miksi, mitä tai miten jokin ilmiö tapahtuu (Eriksson & Koistinen, 2005, s.4–5).

Laineen, Bambergin ja Jokisen (2007) mukaan tapaustutkimus voi olla monenlaista: kriittistä, äärimmäistä, ainutlaatuista, tyypillistä, paljastavaa, tulevaisuutta ennakoivaa tai pitkittäistutkimukseen perustuvaa. Kriittisessä lähestymistavassa pyritään haastamaan, vahvistamaan, tai laajentamaan olemassa olevaa teoriaa. Kriittinen tapaus on juuri se todennäköisin tai epätodennäköisin tapaus. Äärimmäinen tapaustutkimus sopii hyvin kärjistyneiden ilmiöiden tarkasteluun, joissa äärimmäinen tapaus osoittaa ää-

rimmäiset piirteet tapauksesta ja siten osoittaa ne normaalia selkeämmin. Ainutlaatuiset tutkimukset keskittyvät poikkeuksellisiin tai harvinaisiin tapauksiin, joita ei voitaisi johtaa yleisten säännönmukaisuuksien avulla. Tyypillisiä tapauksia analysoimalla voidaan syventää tai laajentaa teoreettista ymmärrystä. Tyypilliset tapaukset oletetaan tutkittavilta seikoiltaan keskimääräisiksi. Paljastava tapaustutkimus tarkastelee ilmiötä, joka tunnetaan, mutta jota ei ole vielä tutkittu. Tulevaisuuteen suuntautuvassa tapaustutkimuksessa tarkastellaan nykyhetkeä ennakoiden tulevaa. Esimerkkinä voidaan käyttää vastaavaa tapausta jollain toisella alalla ja olettaa, että tutkittava tapaus tulee käyttäytymään vastaavasti. Pitkittäistutkimuksessa samaa tapausta seurataan useassa ajankohdassa ja vertaillaan näissä tapahtuneita muutoksia. Haasteena kuitenkin nähdään muiden muuttujien muutos samassa paikassa, mutta eri ajassa. (Laine, Bamberg & Jokinen, 2007, s. 32–34.)

Tässä työssä käytetään laadullista tapaustutkimusmetodia ja pyritään tutkimaan miten erilaiset palvelumallit soveltuvat avoimen lähdekoodin kaupallistamiseen esimerkkien ja toimivien mallien avulla.

### 1.3 Tutkimuskysymys

Avoimen lähdekoodin tuotteita on suhteellisen helppo kopioida, joten ongelman ratkaisua lähdettiin hakemaan eri liiketoimintamallien avulla. Lisäksi lisenssien valinnalla voidaan omaa kehitystä suojata.

Tutkimuskysymykseksi muodostui: Miten avoimen lähdekoodin käyttö vaikuttaa eri liiketoimintamalleissa?

**Taulukko 1 Tutkimuskysymys**

Tutkimuskysymys	Aineisto	Aineiston kerääminen ja käsittely
Miten avoimen lähdekoodin käyttö vaikuttaa eri liiketoimintamalleissa?	Kirjallisuus, artikkelit ja tapausyrityksen kokemukset.	Kirjallisuuskatsaus ajankohdaisiin artikkeleihin ja kirjallisuuteen, sekä tulosten analysointi. Lisäksi tapausyrityksen käytännönkokemukset.

## 2 Teorettinen viitekehys

Gartnerin (2023) hypekäyrän mukaan autonomiset koneet (autonomous vehicles) ovat juuri nyt tulossa yleiseen tietoisuuteen ja ovat saamassa uskottavuutta sitä kautta. Tämä on tyypillisesti juuri se aika, kun markkinoille kannattaa tulla toimivan tuotteen kanssa, koska asiakkailta alkaa olla tietoisuus toimivista ratkaisuista ja he alkavat miettiä vastaavia ratkaisuja omiin tarkoituksiinsa. Tämä tukee helposti toiminnan aloitusta, kun ihan kaikki ratkaisut eivät ole täysin uusia asiakkaille. Hypekäyrän alussa tyypillisesti ollaan vielä niin alussa kehitystyössä, ettei tekniikka oikeasti ole vielä valmis kaupallistettavaksi.

### 2.1 Lean startup metodi

Lean startup metodilla tarkoitetaan metodia, jossa tuodaan markkinoille innovatiivinen tuote (MVP minimum valuable product) mahdollisimman pienellä vaivalla toimiva tuote ja jota testataan asiakkailta, saadaan kokemuksia ja palautetta aikaisessa vaiheessa. Lisäksi tavoitteena on löytää asiat, jota asiakkaat oikeasti haluavat. Samalla markkinan kysyntä tulee heti testattua (Ries, 2016, s. 20). Tuotteen pitää kuitenkin olla sen verran valmis, että se tuottaa jo siinä vaiheessa asiakkaalle arvoa. Avoin lähdekoodi ja avoimet laitteistot soveltuvat hyvin juuri lean startup metodiin ketteryytensä ansiosta.

### 2.2 Avoin lähdekoodi

Avoimella lähdekoodilla tarkoitetaan ohjelmistoa, jonka tekijänsä on sallinut yleiseen jakoon, joten kuka tahansa voi käyttää tai muokata sitä omien tarpeidensa mukaan (Hyvönen, 2003, s. 80).

Avoimen lähdekoodin hyödyntäminen tarjoaa laajat mahdollisuudet nopeaan ja innovatiiviseen ohjelmistokehitykseen, erityisesti organisaatioille, joilla ei vielä ole runsaasti resursseja käytettävissä liiketoiminnan alkuvaiheessa. Tämän lisäksi Avoin lähdekoodin ohjelmistoissa esiintyvät valmiit korjaukset virheisiin vähentävät merkittävästi tar-

vetta aloittaa ohjelmiston kehitysprosessi tyhjältä pohjalta. Avoimen lähdekoodin olennainen etu piilee siinä, että sen kehitystyö on tapahtunut monien erilaisten tarpeiden ja käyttötapausten ympärillä. Näin ollen Avoimen lähdekoodin projektit kattavat laajan kirjon teknisiä haasteita ja tarjoavat ratkaisuja, joita saattaa olla vaikea hahmottaa ohjelmiston kehityksen alkuvaiheessa (Koivisto & muut, 2011, s. 77–78).

Avoimen lähdekoodin käytössä ilmenee etuna valmiin tuen tarjoaminen monille erilaisille oheislaitteille ja antureille, sillä monet muut kehittäjät ovat kohdanneet näihin liittyviä tarpeita aikaisemmissa projekteissaan. Tämä valmiiksi integroitu tuki vähentää merkittävästi kehittäjän työmäärää, kun hänen ei ole integroitava ohjelmistoa erikseen eri laitteiden kanssa.

Tiivistetysti avoimen lähdekoodin käyttö ei ainoastaan tarjoa kustannustehokkuutta ja resurssien säästöjä organisaatioille, vaan edistää myös korkealaatuista ohjelmistokehitystä. Tämä tapahtuu hyödyntämällä laajaa kehittäjäyhteisöä, joka tuo mukanaan monipuolista osaamista ja näkemystä eri teknologioista ja käyttötapauksista. Avoimen lähdekoodin käytöllä voi siten olla merkittäviä vaikutuksia organisaation teknologisen kehityksen ja innovatiivisuuden edistäjänä (Duparc & muut, 2022, s. 6). Lin, & Maruping (2022, s. 17) mukaan viisitoista kahdestakymmenestä yksisarvisesta eli parhaiten menestyvistä startupyhtiöistä perustuu avoimeen lähdekoodiin.

Arvonluonti avoimen lähdekoodin sovelluksissa perustuu läpinäkyvyyteen ja tehokkuuteen verrattuna kokonaan uuden rakentamiseen, kun avoimella lähdekoodilla voidaan laajentaa olemassa olevaa koodia omaan käyttöön sopivaksi. Lisäksi loppukäyttäjät saavat pidemmän tuen yhteisön kautta ja parhaassa tapauksessa voi itse muokata sitä haluamukseen (Lakshman, 2012, s. 357). Vuosituhannen lopulla avoimeen lähdekoodiin perustuvan Linuxin ympärille alkoi kehittyä liiketoimintaa, joka ei perustunut ohjelmiston omistamiseen vaan ne olivat ohjelmistoihin perustuvia ratkaisuja, työkaluja ja osaamista, jota yritykset myyvät (Koivisto, Mikkonen & muut. 2011, s. 131).

## 2.3 Avoimet laitteistot

Avoimet laitteistot (open source hardware) mahdollistavat nopeamman kehityksen laitteistoissa ja on ollutkin suosittua erilaisissa kehitysympäristöissä kuten RaspberryPi tai Arduino. Tyypillisesti nämä ovat yhteensopivia avoimen lähdekoodin kanssa, kuten Linuxin. Avoimissa laitteistoissa eri valmistajat voivat kehittää oman paremman version tai kehittää suoraan avointa laitteistoa (Koivisto & muut, s. 19). Lisäksi näitä voidaan yhdistää, kuten Arduino ja RaspberryPi, jotka toimivat yhdessä. Avointen laitteistojen ansiosta pienet valmistajat voivat tuottaa kehittyneitä laitteita edullisesti ja kustannustehokkaasti, koska heidän ei varsinaisesti tarvitse kehittää laitetta, mutta tarvittaessa kuitenkin voivat myös kehittää laitteistoja edelleen. Tässä kuitenkin tulee sama ongelma, mikä ohjelmistoissa: kehitetty sivuhaara ei ole välttämättä enää yhteensopiva uuden version kanssa.

Ardupilot- ja Pixhawk-alustan kehitys on aloitettu 2009 pienen opiskelijaryhmän toimesta Zürichin teknillisessä yliopistossa (Auterion 2025), mutta vasta viime vuosina on tullut välttämättömiä ominaisuuksia esimerkiksi meidän tarkoituksiimme, kuten esimerkiksi suuren ajoneuvon kalibrointi (large vehicle calibration) joka mahdollistaa kompassin kalibroinnin ilman että koko konetta pyöritetään käsissä. Lisäksi erilaiset automaattiset reitin muodostamismenetelmät ovat tulleet vasta 2023 aikana ohjelmistoon (Github 2025). Pixhawk-ryhmän tavoitteena oli kehittää avoimenlähdekoodin alusta, joka pystyy ohjaamaan erilaisia ajoneuvoja maalla, ilmassa ja vedessä. Myöhemmin siitä kehittyi kaksi eri lisenssillä toimivaa alustaa, jotka ovat Ardupilot, joka toimii GPLv3 lisenssin alla ja PX4, joka toimii BSD lisenssin alla. Lisäksi GPLv3 lisenssiä voidaan käyttää, muokata ja uudelleen julkaista, mutta sitä ei saa sulkea, vaan se on pidettävä avoimena jatkossakin (Ardupilot 2024). Käytännössä BSD-lisenssiä voi muokata ja julkaista omanaan, kunhan alkuperäiset kehittäjät on nimetty dokumentaatioissa. Lisäksi koodi voidaan sulkea, eikä sitä ole mikään pakko julkaista edelleen. (Li-Wen Hsu 2025) Pixhawk-alustan lisenssinä on CC BY-SA 3.0 eli Creative Commons Deed, joka mahdollistaa laitteiston kehityksen, mutta kehitys on jaettava myös muille valmistajille (Creative commons 2025). Käyttöliittymänä käytetään Qground control ohjelmistoa,

joka toimii Android-, Windows-, Linux- tai iOS-laitteilla. Sen lisenssi on niin sanottu kaksoislisenssi, jossa tietyt osat ovat suojattu Apache 2.0 ja osa taas GPLv3 lisensseillä (QGC Quide 2025). Suljetut osat voi ostaa käyttöön vuosimaksulla tai jakaa kaikki koodit, jolloin käyttö on ilmaista. Apache 2.0 lisenssiä voi vapaasti käyttää ja sulkea omaan käyttöön (Apache 2025).

Tässä työssä käytetään esimerkkinä Pixhawk-alustaa, jota tekee useampi valmistaja omilla muokkauksillaan, kuitenkin yhteensopivana Ardupilot- ja PX4 ohjelmistoon. Lisäksi alustaa voi tilata myös kustomoituna versiona omaan tarkoitukseen. Suurin ero näiden alustojen välillä on lisenssi ja sen aiheuttamat erot. Ardupilot kehittyy nopeam-  
paa ja sillä on enemmän käyttäjiä, mutta siitä on vaikeampi tehdä oma tuote, koska oma kehitys täytyy jakaa muillekin.

## **2.4 Palvelumallit asiakkaan ja palveluntarjoajan näkökulmasta**

Palvelullistamisella tarkoitetaan organisaatioiden siirtymistä tuotteiden myynnistä palvelujen ja kokonaisuuksien myyntiin. Palvelullistamisen tutkimus on aloitettu 1980-luvun lopussa (Kohtamäki, Baines, Rabetino & Bigdeli 2018, s.2–3). Palvelullistaminen on liiketoimintamallin kehittämistä tuotekeskeisestä palvelukeskeiseen malliin. Digitalisaatio luo paljon uusia mahdollisuuksia palvelullistamiseen esimerkiksi tarkan ja reaaliaikaisen datan avulla. Digitalisaation avulla yritykset voivat laajentaa asiakaskuntaa ja päästä uusille markkinoille (Kohtamäki & muut 2018, s.29).

Palvelullistamisen tavoitteena on syventää asiakassuhdetta, tyydyttää asiakkaan tarpeita laajemmin ja syvemmin sekä parantaa kilpailukykyä. Yksinkertaisimmillaan yritys voi alkaa tarjoamaan tuotteen ympärille kehitettyjä palveluita kuten koulutus-, huolto-, rahoitus- ja jälkimarkkinointipalveluita. Parhaimmillaan palvelullistaminen voi tarkoittaa koko liiketoiminnan uudistamista, jolloin asiakkaille tarjotaan tuotteen käyttöön perustuvia palveluita, eikä enää tuotteiden omistajuuteen liittyvää myyntiä olisi.

Vastaavasti tuotokeskeisessä ajattelussa tuote on pyritty myymään mahdollisimman monelle asiakkaalle mahdollisimman usein. Tuotteen liian pitkä elinkaari on tässä mallissa yritykselle haitallinen, toisin kuin palvelukeskeisessä mallissa (Kohtamäki & muut 2018, s.17). Tuotteen houkuttelevuutta voidaan nostaa kuukausimaksullisuudella, kuten esimerkiksi vartiointilaitteisto, joka maksaa 1000 € verrattuna kuukausimaksulliseen palveluun, joka maksaa 45 € kuukaudessa. Tämä edellä mainittu palvelumalli tuottaa jo alle kahdessa vuodessa enemmän tarjoajalle kuin kertamaksullinen tuote (Ilmarinen & Koskela, 2015, luku 7).

#### **2.4.1 Tuotemalli, jossa tuotteilla on rajoitettu tuki**

Tuotteet myydään vain pakollisella tuella varustettuna ilman mitään kustomointia. Tyypillisesti asiakas valitsee itselleen parhaiten sopivan tuotteen ja tyytyy siihen. Tämä on kaikista kevyin tuoteratkaisu. Tyypillisiä tuotokeskeisiä palvelumalleja ovat esimerkiksi kodinkoneet. Ne myydään asiakkaalle takuulla ja mahdollisuudella huoltoon, mutta asiakkaan on itse hoidettava esimerkiksi ylläpitoon liittyvät toimet kuten suodattimien putsaus (Kohtamäki & muut, 2018, s.89).

Tämä malli ei sovi kovin hyvin varsinkaan TUOTE 1 tuotteen ansaintamalliksi. Erityisesti siksi, että ohjain on kustomoitava asiakkaan koneelle sopivaksi ja palvelua on tarkoitus päivittää jatkuvasti. TUOTE 2 soveltuu pääosin tähän malliin, koska tuotetta on tarkoitus myydä tiettyyn tarkoitukseen. Esimerkiksi lisäakkuja tai lisättäviä sensoreita voidaan myydä erikseen.

#### **2.4.2 Asennetut ja tuetut tuotteet**

Tässä mallissa laite tai tuote asennetaan asiakkaalle ja pidetään kunnossa sen elinkaaren ajan. Tässä mallissa voi olla myös ulkopuolisia toimijoita, jotka tarjoavat esimerkiksi

korjauspalvelua. Tässä mallissa suurin ero edelliseen on se, että tuote laitetaan toimi-  
maan asiakkaan ympäristössä, eikä vain toimiteta sitä asiakkaalle (Kohtamäki & muut,  
2018, s89-90). Tällaisia tuotteita ovat esimerkiksi hanat.

Tämä malli kuvaa hyvin TUOTE 1 tuotteen palvelumallia, kuitenkin sillä erotuksella, että  
TUOTE 1 perustuu täysin kuukausimaksuun.

### **2.4.3 Täydentävät palvelut**

Tässä mallissa tuotteen lisäksi tarjotaan konsultointia, tuotekehitystä ja tutkimusta itse  
tuotteen ympärille. Lisäksi voidaan tarjota muotoilupalveluja, diagnosointia, logistiik-  
kapalveluja, hallittuja varastoja ja koulutusta. Ansaintamallina tässä mallissa on tarpeen  
mukaan laskutus riippuen mitä täydentäviä palveluja asiakas haluaa käyttää (Kohtamäki  
& muut, 2018, s.90). Tällaisesta palvelusta esimerkkinä on keittiön kaapistot, jotka ovat  
yleensä mittatilaustyötä ja toimittaja myy lisäpalveluja tuotteen ympärille.

Tätä mallia voisi ajatella myös tuotteen 3 ansaintamallina, jossa itse tuote on jonkun  
toisen tarjoama, mutta tätä tuotetta myydään lisävarusteena tai kehittyneempänä oh-  
jausjärjestelmänä tavallisen tilalle.

### **2.4.4 Tuotteeseen keskittyvät palvelut**

Tässä mallissa ratkaisun suunnittelu, toteutus ja tuki on sisällytetty tuotteen kanssa  
myytäväksi. Tuki on tyypillisesti sopimukseen perustuva, kuten esimerkiksi kiinteähin-  
tainen huolto. Tässäkin mallissa itse tuote on asiakkaan omistuksessa ja asiakas käyttää  
tuotetta, mutta ympärille myytävät palvelut ovat erikseen laskutettavia. Tästä mallista  
käytännön esimerkki voisi olla tulostuspalvelu, jossa maksetaan kuitenkin tarvikkeista  
jokin hinta tulostettua sivua kohti, mutta itse laitteet ovat asiakkaan ja tulostuksen hoi-

taa asiakas itse. Palvelu kuitenkin huolehtii, että tulostus onnistuu aina kun asiakas niin haluaa (Kohtamäki & muut 2018, s.90–91).

#### **2.4.5 Leasing**

Leasing eroaa edellisestä pääasiassa siten, että omistus ei siirry asiakkaalle. Tyypillisesti kuukausihintaan on sisällytetty myös tuotteen hinta, joten asiakas maksaa tuotteen ja rahoituspalvelun, mutta omistus ei siirry. Tästä tyypillinen esimerkki on leasing-auto, jonka kilometrit on rajoitettu, mutta asiakkaalle tarjotaan auto tiettyyn kuukausihintaan tietyillä kilometreillä ilman mitään muita kuluja kuin polttoaine. Lisäksi leasingin lopussa saattaa olla lunastushinta, jolla auton voi ostaa itselleen (Kohtamäki & muut 2018, s.91).

#### **2.4.6 Käyttöpalvelu**

Tässä palvelumallissa tuote on asiakkaan, mutta toimittaja tarjoaa käytön palveluna, jossa toimittaja huolehtii käytöstä, huollosta ja ylläpidosta. Tällaisia palvelumalleja käytetään esimerkiksi voimalaitoksissa. Ansaintamallina voi olla esimerkiksi kuukausimaksu, joka sisältää ylläpidon, huollon ja käytön kustannukset. Tyypillisesti käyttöpalvelusopimus tehdään vuosiksi eteenpäin (Kohtamäki & muut 2018, s.91–92).

Tätä voisi soveltaa molempiin tuotteisiin. Olisi selvempää, jos laitteet olisivat asiakkaan omia ja vastuu niiden asianmukaisesta käytöstä olisi selvempi. Samalla kuitenkin palvelun helppous ja arvon luonnin selittäminen tulisi vaikeammaksi asiakkaalle.

#### **2.4.7 Hallittu palveluratkaisu**

Tämä malli eroaa edellisestä pääasiassa siinä, että ansaintamallina voi olla esimerkiksi tietty hinta ulos tuotettua yksikköä kohden. Lisäksi tuote voi olla ulkoistettuna ulkoisel-

le toimijalle. Tässä palvelumallissa on enemmän merkitystä palveluntarjoajalle tuloksella kuin edellisessä (Kohtamäki & muut 2018, s.92).

#### **2.4.8 Kokonaisratkaisu**

Tämä palvelumalli on kaikista laajin ja kattavin. Tässä mallissa toimittaja toimittaa laitteet, huolehtii, että laite toimii ja on huollettu sekä huolehtii käytöstä. Ansaintamallina tässä palvelumallissa on ulostuotettu yksikkö. Esimerkiksi eräs paineilmakompressorivalmistaja on kehittänyt mallin, jossa asiakas saa vanhan kompressorin tilalle tuotettua paineilmaa, jonka tarjoaja hoitaa kokonaisvaltaisesti asiakkaalle tiettyyn kuukausihintaan riippuen tuotetusta kokonaisilmamäärästä (Kohtamäki & muut 2018, s.92–93).

Tätä mallia voisi soveltaa käyttötuntien laskutuksena ja sinänsä se soveltuisi kaikkiin tuotteisiin. Ongelmana näen sen, ettei se kannusta käyttämään tuotetta niin paljon kuin kuukausimaksu.

### **3 Tapaustutkimuskohteen tapauksen ratkaisu**

Tutkimuksessa päädyttiin käyttämään kolmea eri tuotetta case-tapauksina ja niiden tulosten vertailua, koska nämä tuotteet olivat kaikki vähän erilaisia, erilaisille asiakkaille. Kuitenkin kaikissa näissä tuotteissa on käytössä sama ohjelmisto ja osittain myös sama laitteisto.

Laitteisto on niin sanottu kyberfyysinen järjestelmä (CPS), jossa laitteisto pystyy erilaisen anturidatan perusteella tekemään päätöksiä fyysisessä maailmassa, kuten esimerkiksi kääntämään ohjaussyylinteriä tietyn asteluvun tai laskemaan moottorin kierroslukua kääntymiseen tarvittavan määrän Järvinen ja muut, s. 983–990, (2017).

#### **3.1 TUOTE 1 Automaattiohjaus jyrään**

Tässä case-tapauksessa ajatus on korvata kuljettaja erilaisissa työkoneissa ainakin osittain ja tarjota ohjain palveluna. Tässä mallissa arvoa luodaan montaa eri kautta, kuten työtyytyväisyys, työterveys, tarkempi työnjälki, vähemmän päällekkäistä ajoa ja pidemmät työajat koneelle. Lisäksi asiakas saa säästöjä palkkakustannuksissa. Ensimmäinen case mihin ohjainta olisi tarkoitus käyttää on erilaiset jyrät, joilla työskentely on tyypillisesti monotonista, toistuvaa ja hidasta. Lisäksi jyrän kyydissä istuminen altistaa tärinälle. Koska tekniikka ei vielä toimi 100 % varmuudella, ei jyrää voida jättää valvomatta työskentelemään, mutta operaattori tai vastuullinen voi kuitenkin tehdä samalla jotain muuta. Tyypillisesti tällaisilla työmailla on kaivinkoneen kuljettaja, joka voi samalla valvoa useampaa jyrää. Edellä mainittu järjestely säästäisi merkittävästi palkkakustannuksissa. Sen lisäksi työn jälki on aina tasalaatuinen, eikä kuljettaja kyllästy niin helposti. Ohjaimessa kaikki komponentit ovat helposti vaihdettavissa ja siihen päästään helposti etänä käsiksi. Tukipalvelu on suhteellisen helppo järjestää. Lisäksi yksittäiset osat ovat halpoja, joten niitä voi olla vaikka jokaisella asiakkaalla varastossa.



**Kuva 1 Tuote 1 Autonominen jyrä**

Tässä case-tapauksessa päädyttiin juuri palvelumalliin sen vuoksi, ettei ohjaimesta voisi kovin helposti pyytää riittävän suurta summaa ja samalla kuitenkin tuki olisi myytävä lisäpalveluna. Koska pohjalla on avoin lähdekoodi, se päivittyy noin kolmen kuukauden välein ja kaikki asiakkaiden laitteet on tarkoitus pitää ajan tasalla jatkuvasti. Tämä helpottaa myös tukea merkittävästi, koska käytännössä jokainen ohjain tulee olemaan samanlainen ja käyttää samaa ohjelmistoversiota. Tämän lisäksi palvelumalli, jossa aina uusien ohjelmistojen sitouttaa asiakasta ja nostaa kynnyksen vaihtaa mihinkään vaihtoehtoiseen ratkaisuun. Asiakaskartoituksessa selvisi, että asiakkaat haluavat nimenomaan palvelumallin, eikä kertamaksullista tuotetta. Yhtenä ongelmana jyräntapauksessa on työn kausiluonteisuus, joka ajoittuu pääasiassa kesään. Tuote kuitenkin tuottaa arvoa jo sesongin aikana enemmän kuin koko vuoden palvelumaksut olisivat. Hinnoittelussa on lisäksi asennuskustannus ja purkukustannus, joiden on tarkoitus houkuttaa jatkuvampiin sopimuksiin.

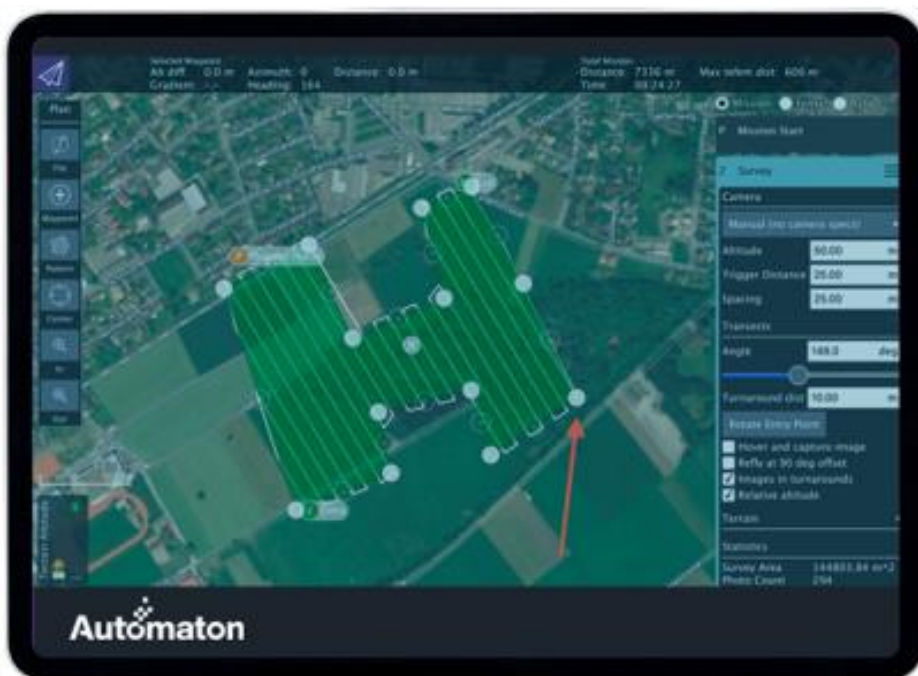
### 3.2 TUOTE 2 Kuusikopteri automaattiajolla

Toinen tuote mitä olemme kehittäneet ja testanneet on kuusikopteri (hexacopter), joka pystyy kantamaan noin 25 kg kuorman. Tätä tuotetta on käytetty lähinnä piensiemenkylvöpalveluna, koska lentäminen on vaikeaa ja tuotteen käyttö vaatisi operaattorilta paljon osaamista ja lisäksi Traficomien hallinnoiman luvan. Palvelu on hinnoiteltu hehtaarihinnoittelulla, koska työ on erikoistyötä, mitä ei voi suorittaa muuten kuin lentämällä. Hehtaarin kylvö kestää noin 7 minuuttia, joten laite on melko tehokas verrattuna muihin menetelmiin. Tyypillisesti kuitenkin asiakkaat ovat halunneet palvelun pelloille, joissa on märkää tai muuta kasvustoa. Käyttämällä tätä palvelua, oleellinen etu on, että ei siitä jää jälkiä tai muuten vahingoita peltoa tai muuta kasvustoa. Esimerkiksi kerääjäkasvin kylvö ei käytännössä onnistu muuten kuin viljan kylvön yhteydessä tai tällä kuusikopterilla kasvuston sekaan. Teknisesti kuusikopterissa on 14 litran säiliö ja levittäjä, jota kontrolloi Pixhawk-alusta.



Kuva 2 Tuote 2 Autonominen kuusikopteri

Android tabletilla piirretään pellon reunat ilmapuuvuon ja määritellään työleveys, tehtävä lähetetään radiolinkin kautta kuusikopteriin. Lentoon voidaan puuttua milloin tahansa ja operaattorin on syytä olla jatkuvasti valmiina ottamaan kontrolli. Lisäksi akkuja ja siemeniä pitää täyttää suhteellisen usein. Kuusikopteri osaa tulla hakemaan lisää siemeniä ja laskeutua itsestään, jos akun jännite laskee alle tietyn rajan. Tulevaisuudessa tätä tuotetta voitaisiin myydä myös tuotteena suoraa asiakkaalle, kunhan saadaan se riittävän varmaksi ja helppokäyttöiseksi. Yksi haaste oli esimerkiksi siementen levitysmäärä, joka muuttuu ilmankosteuden, siemenlajien ja siementen kosteuden mukaan. Levitysnopeus pitää siis synkronoida joka päivä uudestaan.



Kuva 3 Käyttöliittymä taulutietokoneella

### 3.3 TUOTE 3 Autonominen saunalautta

Kolmas tuote on autopilot-järjestelmä vesillä liikkuviin aluksiin tai veneisiin. Tätä tuotetta myydään tuotemallin avulla suoraa valmistajille, koska palvelumallina tämä olisi hankala myydä. Tuotteesta on kolme erilaista variaatiota, joista helpoin on kahdella tai useammalla moottorilla tehty liuku-ohjaus, jossa moottorien nopeutta säätämällä voidaan muuttaa suuntaa.



**Kuva 4 Tuote 3 Autonominen saunalautta**

Tässä autopilotin mukana tulee kauko-ohjaus ja sähköohjaus, joten mitään mekaanista ohjausta ei enää ole käytössä. Toisessa variaatiossa servomoottori kääntää perä- tai etumoottoria, jolloin suuntaa voidaan muuttaa tarkasti ja nopeasti. Kolmannessa variantissa ohjaus tapahtuu sähkösylinterin avulla kääntämällä perämoottoria tai peräsintä. Lisäksi näitä eri ohjauksia voidaan yhdistää. Riippumatta ohjaustavasta, käytetään samaa Ardupilot-ohjelmistoa, jota on esitelty aiemmin. Valmistajille myytäessä voidaan

ohjelmistoa yksilöidä hieman valmistajaa varten lisäämällä esimerkiksi logot tai jotain tiettyjä pikanappeja.

### **3.3.1 Turvallisuus**

Suurin huoli asiakaskartoituksen mukaan oli laitteen turvallisuus, joka nousi useimmiten ongelmaksi. Tätä ongelmaa on yritetty ratkaista sillä, että tyypillisesti alue on rajattu joko aidoin tai sitten koko alue on niin kaukana asutuksesta, ettei kukaan sinne vahingossa joudu. Lisäksi turvallisuudesta huolehtii lidar, joka pysäyttää jyrän, jos jokin tulee sen tielle. Tässä tulee ongelmaksi esimerkiksi makaava henkilö, jota lidar ei näe. Siihen ratkaisuksi 3d-kamera on tarkoitus saada tunnistamaan objekteja ja antamaan pysäytys- tai väistökäskyn laitteelle. Tämän lisäksi valvojalla on velvollisuus valvoa, ettei alueelle tai koneen tielle tule mitään. Lidarilla tarkoitetaan laser etäisyysmittaukseen perustuvaa sensoria, joka voi mitata etäisyyksiä tarkasti. 3d-kameralla tarkoitetaan tuplakameraa, joka toimii saman tyyllisesti kuin ihmisen syvyysnäkö. Siinä voidaan lisäksi käyttää edellä mainittua lidaria varmistamaan etäisyydet. Kuitenkin ongelmaksi muodostuu toimintavarmuus, mitä enemmän erilaisia sensoreita tai varmistuksia lisää, sitä enemmän erilaisia ongelmia ne aiheuttavat. Samalla kuitenkin erilaisten toimintojen lisääminen lisää myös lisäarvoa asiakkaalle ja tekee tuotteesta arvokkaamman. Lisäksi erilaiset toiminnot tekevät kilpailijoiden kopioinnin vaikeammaksi. Tähän voidaan soveltaa Michael Porterin viiden kilpailukyvyyn mallia, jossa kilpailijan on hankalampi tulla markkinoille, kun kilpailukykyinen tuote on jo olemassa (Porter, 1985, s. 23–26). Riippumatta siitä yleistyykö tämä palvelu, autonomiset koneet tulevat jossain vaiheessa yleistymään joka tapauksessa. Kuitenkin koneiden käyttöikä on niin pitkä, että jälki-asennettava palvelu tulee olemaan tarpeellinen. Lisäksi ohjainta voidaan soveltaa melkein mihin tahansa koneeseen, jyrä on vasta ensimmäinen askel.

### **3.3.2 3D-kamera OAK depthAi**

Tässä case-tapauksessa on käytetty PX4-alustan lisäksi OAK depthAi-kameraa, jossa on sisäänrakennettuna tekoälyyn tarvittava prosessori ja 3d-kamera. Kameraa on tarkoitus

käyttää objektin tunnistamiseen ja 3d-kameran avulla sen etäisyyden määrittämiseen. Objekteja varten on olemassa valmiita kirjastoja kuten opencv ja TensorFlow. Sen lisäksi voidaan käyttää esimerkiksi Roboflow-työkalua, jolla on helppo opettaa itse objekteja. Lisäksi palvelussa on muiden opettamia objektitietokantoja. DepthAi on alusta asti ollut yhteensopiva Ardupilot-ohjelmiston kanssa ja perustuu myös avoimeen lähdekoodiin. Erityisen hyödyllinen 3d-kamera on erilaisten korkeuserojen ja esimerkiksi oijen tunnistamiseen. Kun se olisi erikoisen hankalaa esimerkiksi lidarin avulla (Luxonis 2025).

### **3.3.3 Kilpaileva alusta Ros2**

Kilpaileva alusta Ros2 on avoimen lähdekoodin robotiikka-alusta, joka tarjoaa laajan valikoiman työkaluja robottien hallintaan ja käyttöön. Siitä on tullut yhä suosittu vaihtoehto robotiikassa, ja sen avulla on kehitetty lukuisia kaupallisia sovelluksia. Se mahdollistaa robottien hallinnan ja kommunikaation, mikä parantaa tehokkuutta ja vähentää virheitä. Ros2 on tärkeä sovelluskohde robotiikan opetuksessa ja tutkimuksessa. Sen avoimen lähdekoodin luonne mahdollistaa opiskelijoiden ja tutkijoiden kehittämisen ja testata uusia robotiikkakäyttötapoja ja ratkaisuja helposti. Ros2 alustaa käytetään yleisesti monissa teollisissa ratkaisuissa, kuten Caterpillarin pusku- ja traktoreissa (Ros industrial 2025).

## 4 Tulokset

Eri liiketoimintamallit soveltuvat eri tuotteilta eri lailla riippuen asiakkaan tarpeista. Tulokset valituille tuotteille näillä kahdella liiketoimintamallilla eli tuotemalli, jossa tuotteen omistaa asiakas ja kokonaispalvelu, jossa tuote on palveluntarjoajan ja asiakas vain tilaa tuotetta kuukausitilauksena. Tuote 1 soveltuu paremmin kokonaispalvelun avulla, koska tuella on suurempi merkitys tuottavuuteen ja tuote tuottaa parhaiten arvoa juuri palveluna. Lisäksi kuukausihinnan ollessa edullinen, kynnyks lähteä kehittämään omaa kilpailevaa versiota kasvaa. Tuotteen 2 kanssa tuotemalli soveltuu paremmin, koska tuote on enemmän suunnattu erikoisammattilaisten käyttöön. Lisäksi tuotteen käyttäjän vastuut ovat selkeämmät, jos tuote on asiakkaan omistuksessa. Tuotteen 3 liiketoimintamalliksi soveltuu paremmin tuotemalli, koska sitä myydään suoraan valmistajille.

Taulukko 2 Tulokset tuotteelle 1

Tuote	Liiketoimintamalli	Edut	Haitat	Sitoutuminen	Arvo
<b>TUOTE 1</b>	Tuotemalli	Vastuut tuotteen käytöstä tuotteen ostajalla. Vastuu tuotteen toiminnasta vain takuuajan.	Myynti hankalampaa	Yhteen tuotteeseen kerrollaan. Ylläpito ja päivitykset maksullista.	Tuottaa arvoa koko elinkaaren ajan.
<b>TUOTE 1</b>	Kokonaispalvelu	Myynti helpompaa matalalla kuukausihinnalla. Jatkuva kassavirta.	Jatkuvan ylläpidon vastuu.	Kuukausihinta kattaa ylläpidon ja päivitykset, joten asiakas sitoutuu palvelun ajaksi.	Tuottaa arvoa toistaiseksi

Taulukko 3 Tulokset tuotteelle 2

Tuote	Liiketoimintamalli	Edut	Haitat	Sitoutuminen	Arvo
<b>TUOTE 2</b>	Tuotemalli	Vastuut tuotteen käytöstä tuotteen omistajalla.	Tulot yksittäisiä, päivittävyys jää asiakkaan huoleksi.	Asiakas voi helposti ostaa eri tuotteen seuraavaksi, eikä mikään sido.	Tuottaa arvoa koko elinkaaren ajan
<b>TUOTE 2</b>	Kokonaispalvelu	Hankala myydä kuukausimaksullisena, lisäksi hinnoittelu riippuisi käyttötarkoituksesta.	Vikaherkkyys, hankala saada tuote pysymään kunnossa, jos asiakas ei tunne, että se on oma.	Helppo lopettaa palvelusopimus	Tuottaa arvoa koko sopimuskauden ajan.

Taulukko 4 Tulokset tuotteelle 3

Tuote	Liiketoimintamalli	Edut	Haitat	Sitoutuminen	Arvo
<b>TUOTE 3</b>	Tuotemalli	Vastuut tuotteen käytöstä tuotteen omistajalla. Valmistaja voi myydä asiakkailleen varusteena.	Tulot yksittäisiä, päivitettävyyttä jää asiakkaan huoleksi.	Asiakas voi helposti ostaa eri tuotteen seuraavaksi, eikä mikään sido.	Tuottaa arvoa koko elinkaaren ajan.
<b>TUOTE 3</b>	Kokonaispalvelu	Hankala myydä kuukausimaksullisena, lisäksi hinnoittelu riippuisi käyttötarkoituksesta.	Asiakkaan pitäisi ostaa tukea takuuajan jälkeen erikseen.	Helppo lopettaa palvelusopimus.	Tuottaa arvoa koko sopimuskauden ajan.

## 5 Diskussio

### 5.1 Johtopäätökset

Tutkimuksen perusteella voidaan todeta, että tietynlaisilla avoimeen lähdekoodiin perustuvilla tuotteilla palvelumalli on paras tapa suojata tuotetta muilta toimijoilta. Tämän case-tapauksen jyrä soveltuu erityisen hyvin palvelumalliin, eikä tuotemalliin. Kuitenkin esimerkiksi erikseen myytäviä tuotteita, kuten lentäviä laitteita on parempi suojata oman sivuhaaran kautta ja tehdä koodista siten suljettua.

Tuotetta 2 on järkevämpää myydä tuotemallin avulla, koska vastuut käytöstä ja hallinnasta on selkeämpää jättää käyttäjän harteille. Lentävien laitteiden myynti palveluna ei olisi kovin helposti toteutettavissa, eikä välttämättä tuottaisi haluttua arvoa asiakkaalle. Lisäksi palvelun tarjonta rajoittuu tietyille alueille tai pitäisi hakea erillisiä operaattoreita eri palvelualueille. Hankaluuksia tuottaisi esimerkiksi operaattorin tekemät virheet ja vastuut yleisesti.

Tuotteessa 3 taas tuotemalli soveltuu paremmin, koska tuotetta myydään valmistajalle, eikä käyttäjälle. Valmistajan on selkeämpää myydä tuotetta lisävarusteena, eikä palveluna. Tuotteen 3 kohdalla avoimen lähdekoodin lisenssit kannattaa sulkea sen jälkeen, kun on riittävästi resursseja kehittää lisäominaisuuksia.

Palvelun tavoitteena on tuottaa mahdollisimman paljon arvoa asiakkaalle. Gröönroos (2015 s. 195) mukaan arvon tuottaminen onnistuu helpoiten juuri palvelumallin avulla, kun voidaan jatkuvasti varmistaa, että asiakkaalla on uusin ja toimiva ohjelmisto käytössään. Se että laskutus on selkeä kuukausiveloitus, tekee myös asiakkaalle siitä helppoa ja vaivatonta. Jatkuva tuki ja ylläpito antaa myös asiakkaalle tunteen, että hänen laitteestaan huolehditaan. Lisäksi jatkuva kassavirta tuntuu palveluntuottajalle paremmalta, kuin jälkimarkkinointi pakollisena hallinnollisena palveluna.

Ries (2016 s. 93) mukaan (MVP) A minimum viable product eli mahdollisimman pienellä vaivalla rakennettu toimiva tuote on tärkeää saada mahdollisimman nopeaa tuottamaan kokemusta, asiakaspalautetta ja oppia mihin suuntaan tuotetta kannattaa viedä.

MVP on helpoin ja nopein saada juuri avoimen lähdekoodin ja avointen laitteistojen avulla toimimaan. Lisäksi sen avulla ensimmäiset asiakkaat pääsevät vaikuttamaan tu-leviin ominaisuuksiin.

Käytännön suosituksina voidaan pitää tutkimuksen perusteella, että aluksi kannattaa käyttää ja kehittää laitteet toimimaan Ardupilot-ohjelmistolla, kunnes on resursseja ja kokemusta omia muokkauksia varten. Kun niitä on kerrytetty tarpeeksi, kannattaa siirtyä PX4-ohjelmistoon ja sitä kautta alkaa kehittää omaa sivuhaaraa ja sulkea koodi. Tällä tavoin voidaan suojata omaa työtä ja kehittää vapaammin ohjelmistoa. Näiden alus-tojen lisenssierot tekevät valinnasta selvän, koska Ardupilot-ohjelmiston lisenssi ei salli koodin sulkemista omiin tarkoituksiin, mutta PX4 taas sallii. Mahdollisen oman laitteis-ton kehittäminen on tehtävä avoimena, mutta samalla, kun tarpeet ovat yksilölliset, alusta ei välttämättä sovellu muiden käyttöön, joten sen suojaamista ei erikseen suosi-tella. Käyttöliittymän kehitys kannattaa aluksi tehdä avoimella lisenssillä, mutta kun omalla käyttöliittymällä alkaa olla arvoa, se kannattaa sulkea ja ostaa vuosilisenssi sitä varten.

Tulevaisuudessa, kun tulee lisää kokemuksia eri palvelumalleista, voidaan tutkia sy-vemmin tuloksia ja eri valintojen vaikutuksia. Lisätutkimusta voisi saada muilta vastaa-vilta toimijoilta ja ROS2 alustaa käyttäviltä yrityksiltä, jotka miettivät samoja avoimen lähdekoodin ongelmia.

## Lähteet

Aarnos, E., & Valli, R. (2018). *Ikkunoita tutkimusmetodeihin: 1, Metodien valinta ja aineistonkeruu : virikkeitä aloittelevalle tutkijalle* (5., uudistettu painos.). PS-kustannus.

Anderson, C. (2012). *Makers: The new industrial revolution*. Crown Business.

Apache. (2025) Apache License, Version 2.0. Noudettu 4.2025 osoitteesta <https://www.apache.org/licenses/LICENSE-2.0>

Ardupilot (2024). License (GPLv3) Noudettu 04.2025 osoitteesta: <https://ardupilot.org/dev/docs/license-gplv3.html>

Auterion. (2025) The story of PX4 and Pixhawk. Noudettu 4.2025 osoitteesta <https://auterion.com/company/the-history-of-pixhawk/>

Bock, A. J., & George, G. (2018). *The business model book: Design, build and adapt business ideas that thrive* (First edition.). Pearson.

Cennamo, C., Dagnino, G. B., & Zhu, F. (2023). *Research handbook on digital strategy*. Edward Elgar Publishing.

Creative commons (2024). Attribution-ShareAlike 3.0 Unported. Noudettu 04.2025 osoitteesta: <https://creativecommons.org/licenses/by-sa/3.0/legalcode>

Duparc, E., Möller, F., Jussen, I., Stachon, M., Algac, S., & Otto, B. (2022). Archetypes of open-source business models. *Electronic markets*, 32(2), 727-745. <https://doi.org/10.1007/s12525-022-00557-9>

- Garner (2023). What's New in Artificial Intelligence from the 2023 Gartner Hype Cycle. Noudettu 04.2025 osoitteesta: <https://www.gartner.com/en/articles/what-s-new-in-artificial-intelligence-from-the-2023-gartner-hype-cycle>
- Github (2025). Rover Release Notes. Noudettu 04.2025 osoitteesta: <https://github.com/ArduPilot/ardupilot/blob/master/Rover/ReleaseNotes.txt>
- Grönroos, C., & Tillman, M. (2015). *Palvelujen johtaminen ja markkinointi* (5. painos.). Talentum.
- Eriksson, P. & Koistinen, K. (2005). Monenlainen tapaustutkimus. Kuluttajatutkimuskeskus. Julkaisuja 4:2005. Kerava: Savion Kirjapaino Oy.
- Hyvönen, E., Kuusisto, T., & Werner Söderström osakeyhtiö. (2003). *Ohjelmistoliiketoiminta* (1. p.). WSOY.
- Ilmarinen, V. & Koskela, K., (2015). *DIGITALISAATIO - Yritysjohdon käsikirja*. Alma Talent.
- Kohtamäki, M., Baines, T., Rabetino, R. & Bigdeli, A. (2018). *Practices and Tools for Servitization: Managing Service Transition*. Palgrave Macmillan US.
- Koivisto, M., Säynäjäkangas, J. & Forsberg, S., (2019). *Palvelumuotoilun bisneskirja*. Alma Talent.
- Koivisto, T., Mikkonen, T., Vadén, T., Valkokari, K., Ahonen, M., & Vainio, N. (2011). *Rajoja ylittävä innovointi*. Tampere University Press.
- Laine, M., Bamberg, J. & Jokinen, P. (toim.). (2007). *Tapaustutkimuksen taito*. Helsinki: Yliopistopaino.

- Lakshman, (2012). Open source data warehousing and business intelligence Taylor & Francis Group <https://ebookcentral-proquest-com.proxy.uwasa.fi/lib/tritonia-ebooks/reader.action?docID=989830&ppg=28> S.357
- Li-Wen Hsu (2025). The FreeBSD Copyright. Noudettu 04.2025 osoitteesta: <https://www.freebsd.org/copyright/freebsd-license/>
- Luxonis (2025). OAK-D S2. Noudettu 04.2025 osoitteesta <https://docs.luxonis.com/projects/hardware/en/latest/pages/DM9098s2/>
- Lin, Y., & Maruping, L. M. (2022). Open Source Collaboration in Digital Entrepreneurship. *Organization science (Providence, R.I.)*, 33(1), 212-230. <https://doi.org/10.1287/orsc.2021.1538>
- Järvinen, T., Loritea, G., Rautioa, A., Juhász, K., Kukovecz, Å., Kónya, Z., Krisztian Kordas, K. & Toth, G (2017). *Portable cyber-physical system for indoor and outdoor gas sensing*, *Sens. Actuators B Chem.*, vol. 252, Nov. 2017, <https://doi.org/10.1016/j.snb.2017.06.102>.
- Porter, M. E., Tillman, M., & Tillman, M. (1985). *Kilpailuetu: Miten ylivoimainen osaaminen luodaan ja säilytetään*. Weilin + Göös.
- QGC guide (2025). QGroundControl License Noudettu 04.2025 osoitteesta <https://docs.qgroundcontrol.com/master/en/qgc-dev-guide/contribute/licences.html>
- Ries, E., Rautanen, A., & Markula, J. (2016). *Lean startup: Kokeilukulttuurin käsikirja : kuinka jatkuvan innovoinnin avulla luodaan merkittävästi onnistuneempaa liiketoimintaa*. LavasDesign Oy.

Ros Industrial (2025). Ros Industrial. Noudettu 04.2025 osoitteesta:  
<https://rosindustrial.org/current-members>

Saarela-Kinnunen, M. & Eskola, J. (2010). Tapaus ja tutkimus = Tapaustutkimus? Teoksessa Aaltola, J. & Valli, R. (toim.) Ikkunoita tutkimusmetodeihin 1. Metodien valinta ja aineistonkeruu: virikkeitä aloittelevalle tutkijalle. Juva: WS Bookwell Oy

Valli, R., & Aarnos, E. (2018). *Ikkunoita tutkimusmetodeihin: 1, Metodien valinta ja aineistonkeruu : virikkeitä aloittelevalle tutkijalle* (5., uudistettu painos.). PS-kustannus.