

EFICODE PIKAOPAS

DEVOPS

eficode

WWW.EFICODE.FI

**Devops on ohjelmistokehitysmenetelmä
agilen jälkeiseen maailmaan. Se nojaa automaatioon,
virtualisaatioon ja fiksuihin työkaluvalintoihin. Tylsää
työstä tulee robotin hommaa, ja ihminen saa tehtäväkseen
ohjelmistotuotannon parhaan palan eli luovan
ja asiakaskeskeisen kehitystyön.**

EFICODE PIKAOPAS

Liikkeenjohdolle

SIVUT: 04 - 10

EFICODE PIKAOPAS

Asiantuntijoille

SIVUT: 11 - 18

EFICODE PIKAOPAS

DEVOPS

eficode

Liikkeenjohdolle

**”Kuinka monta asiaa
onkaan hylätty tänään vain tullakseen
todeksi huomenna!”**

Jules Verne (1828 - 1905)

DEVOPS, KOLMANNEN SUKUPOLVEN
OHJELMISTOKEHITYSMENETELMÄ

Devops on osa liike-elämän ja julkisten organisaatioiden maailmanlaajuista muutosta, jossa tietotekniikkaa ja sen infrastruktuuria virtualisoidaan. Konsultointiyhtiö Deloitte nimesi devopsin yhdeksi vuoden 2014 teknologiatrendeistä.¹

EFICODE TUTKITUTTI

66 %

Suomalaispäättäjistä pitää devopsia kiinnostavana yrityksensä näkökulmasta.²

Ohjelmiston kehittäminen alkaa liiketoiminnallisesta tarpeesta. Kiikarissa voi olla vaikkapa kassan kilisemään saava verkkopalvelu tai digitaalinen työkalu sujuvoitamaan oman henkilöstön arkea. Kun idea on muotoutunut ja tavoitteet asetettu, alkaa kehitystyö. Hetken kuluttua kehitystiimi esittelee valmista tuotetta, jonka käyttäjät ottavat hurraten vastaan. Tai sitten ei. Turhan usein budjetit paukkuvat, aikataulut venyvät ja – mikä pahinta – käyttäjät eivät ole tyytyväisiä.

Missä mättää? Kyse on pääsääntöisesti tavasta tehdä ohjelmistoja.

Ensimmäinen valtavirran ohjelmistokehitysmenetelmä tunnetaan nimellä vesiputousmalli. Sen ytimessä ovat tiukasti etukäteen määritellyt ominaisuudet, joiden toteutumista valvotaan pitkään ja hartaasti suunnitellun sopimuksen pohjalta. Kun kaikkea ei osata kirjata paperille etukäteen, on seurauksena tyypillisesti katastrofi käyttäjäkokemuksen suhteen. Budjetti paukkuu ja aikataulu venähtää viimeistään silloin, kun valmistuneen ohjelmiston huomataankin tarvitsevan korjausta.

¹ - Deloitte Development LLC (2014) Tech Trends 2014: Inspiring Disruption. Deloitte University Press.
² - Markkinatutkimustoimisto Value Clinic Oy (2014) DevOps Suomessa - tutkimusraportti.

FAKTAT TISKIIN

18 %

Pienemmät kehitys- & ylläpitokulut
= Kehitys- ja ylläpitokuluissa säästyy miltei viidennes.³

50 %

Vähemmän vikoja
= Tuplasti parempi mahdollisuus onnistua tuotteen julkaisussa.⁴

30x

Nopeampi julkaisu
= Julkaisu 30 kertaa useammin ja 8000 kertaa nopeammin.⁴

20 %

Enemmän bisnestä
= Liikevaihdon ja uusien asiakkaiden määrän kasvu 17–23%.³

³ - CA Technologies (2013) TechInsights Report: What Smart Businesses Know About DevOps. CA Technologies.

⁴ - Puppet Labs (2013) 2013 State of DevOps Report. IT Revolution Press.



Muuttuvien vaatimusten ongelmaa ratkovat ketterät (engl. agile) menetelmät, joissa ohjelmistotuotantoa lähestytään jaksotetun kehitysproessin ja kommunikaation näkökulmasta. Ketterien menetelmien myötä moni asia korjaantuu, mutta konfliktit teknisen ylläpidon kanssa sekoittavat pakkaa. Ohjelmistoja ostavia huolestuttaa myös ketteriin menetelmiin monesti liitetty avoimen piikin mentaliteetti, joka uhkaa paisuttaa projektin budjetin.

Ohjelmistotuotanto virtualisoituu ja automatisoituu

Ketteryyden hyviä puolia lainaa devops, kolmannen sukupolven ohjelmistokehitysmenetelmä, joka nojaa tuotannon virtuaalisuuteen ja automaatioon. Devopsin myötä ohjelmistokehitys menee pilveen, ja manuaaliset toisteiset työvaiheet poistuvat esimerkiksi laadunvarmistuksesta ja ohjelmiston julkaisemisesta.

Kehitystyön tuloksia voidaan esitellä tilaajille ja loppukäyttäjille sitä mukaa, kun niitä saadaan aikaiseksi. Devops mahdollistaa lisäksi järjestelmän vaatimusten ja automaattisen

laadunvarmistuksen yhdistämisen, jolloin jokainen kehittäjästä johtoon pystyy seuraamaan tuotantoprosessin etenemistä reaaliaikaisesti. Tuotekehityksen keskiössä on aina asiakas ja hänen tavoittelemansa hyöty, ei mikään muu.

Devopsin periaatteita hyödyntävä organisaatio testaa ja julkaisee ohjelmistoa jatkuvasti ja automaattisesti. Kehitys tehdään virtualisoidussa ympäristössä. Tämä vähentää virheitä sekä työstä ja infrastruktuurin hallinnasta aiheutuvia kustannuksia. Kehittäjien ja teknisen ylläpidon yhteiset mittarit auttavat tavallisesti omissa poteroissaan viihtyviä ammattilaisia puhaltamaan yhteen hiileen käyttäjäkokemuksen puolesta.

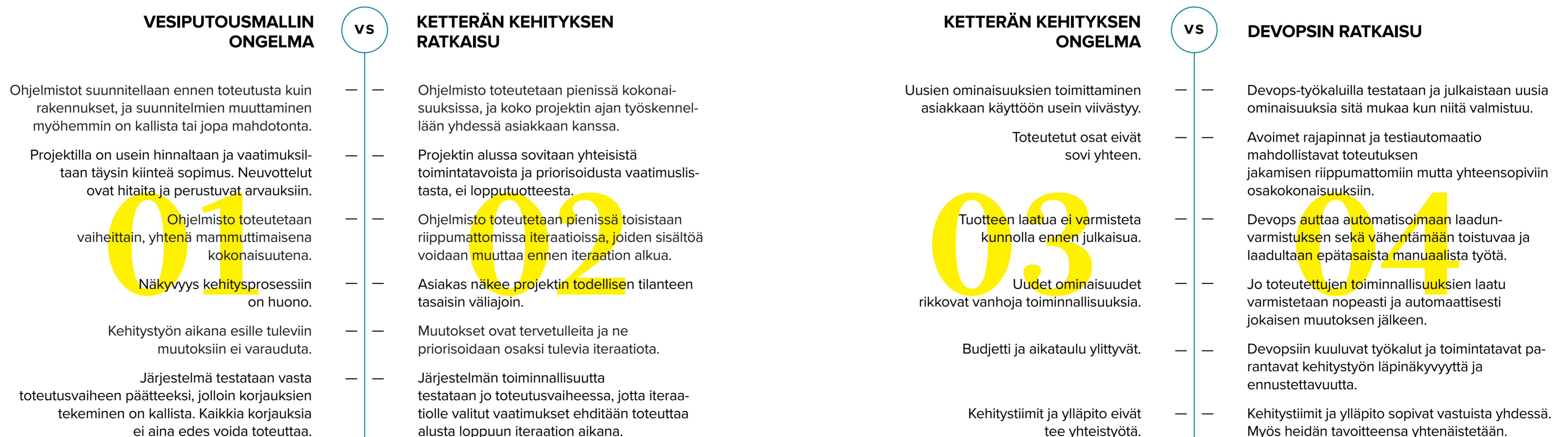
Devops vastaa edeltäjiensä haasteisiin

Ketterä kehitys ratkaisi ison osan vesiputouksellisten ongelmista. Devops jatkaa siitä mihin ketterät menetelmät jäivät ja karistaa samalla tukun agilen huonoja puolia pois.

Eroon tuotantoketjun pullonkauloista

Ketterät menetelmät muuttivat ohjelmistotuotannon käytännöt joustaviksi. Agilessa ei kuitenkaan oteta huomioon it-ympäristöjä ja ohjelmistotuotannossa käytettäviä työkaluja. Tämä synnyttää tuotantoketjun hidastavia pullonkauloja. Devopsissa myös teknologiset ratkaisut ovat tärkeässä roolissa. Esimerkiksi vanhanaikaiset konosaliratkaisut voidaan vaihtaa virtuaalisiin ja prosessit automaattisiin.

ONGELMAT / RATKAISUT



Ei se ihme pilleri ole

Oppimishaluiselle organisaatiolle devopsista on paljon hyötyä, mutta mikään hopealuoti se ei ole. Manuaalinen työ ei katoa devopsin myötä. Sen sijaan toisteinen, puuduttava ja nopeutta vaativa työ siirtyy koneen tehtäväksi. Devopsin myötä kehittäjät voivat keskittyä arvoa tuottavaan työhön eli laatuun ja asiakaskokemukseen.

Tehostamisen alkuun pääsee keveästi

Devops ei vaadi massiivista implementaatioprojektia tai totaalista kulttuurinmuutosta. Päätset alkuun keveästi kartuttamalla ohjelmistotuotantosi tilan ja etsimällä kehitystiimejä vauvaavat kipupisteet.

Tästä löydät johtajan askelmerkit kohti tehokkaampaa tulevaisuutta:

01

Juttele organisaatiosi it-johtajan kanssa ohjelmistotuotantonne tilasta ja devopsin periaatteiden soveltuvuudesta teille. Näytä myös tämän oppaan tekninen puoli hänelle.

**02**

Jos organisaatiostanne ei löydy devopsista innostunutta hankeveturia, etsi sellainen ulkopuolelta. Varmista, että palveluntuottaja ei pelkästään toteuta mekaanisia asennustöitä, vaan on valmis tekemään devopsista organisaatiosi osaamis-pääomaa.

KEHITYSTYÖ**KEHITYSTYÖ PERINTEISISSÄ YMPÄRISTÖISSÄ**

It-ympäristöt ovat joustamattomia ja niiden ylläpito on kallista.

Toimittajariippuvuus pienentää osaamispalettia ja pakottaa turhiin kompromisseihin.

Asiakkaan vaatimusten täyttymistä on vaikea seurata. Julkaisuaikataulun arviointi saattaa olla mahdotonta.

Inhimilliset virheet laadunvarmistuksessa johtavat laatu-ongelmiin.

Tuotekehitystyötä tekevät turhautuvat samanlaisina toistuviin työtehtäviin.

vs

KEHITYSTYÖ DEVOPS-MENETELMÄLLÄ

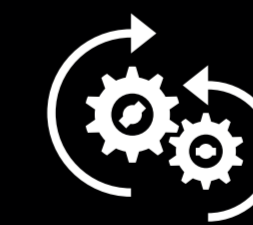
Virtualisoiduilla ympäristöillä ylläpidetään perustoimintoja edullisesti. Samalla varaudutaan myös kuormituspiikkeihin.

Devopsin periaatteet tukevat modulaarisuutta ja avoimia rajapintoja. Yhden toimittajan loukkuun ei tarvitse jäädä.

Devopsin avulla asiakkaan vaatimukset voidaan keskittää yhteen paikkaan ja liittää laadunvarmistukseen. Tieto ominaisuuden julkaisusta tavoittaa asiakkaan heti.

Testiautomaatio hoitaa laadunvarmistuksen aina täydellä teholla.

Koneet tekevät tylsät työt, ja ihmiset pääsevät tekemään luovaa työtä.



DEVOPS SOPII MUUTOSHALUISILLE



Devops sopii organisaatioille, jotka:

01. kehittävät, ylläpitävät tai ostavat ohjelmistoja
02. haluavat kehittää parempia digitaalisia palveluja
03. etsivät kustannussäästöjä
04. tahtovat kehittää toimintaansa.

LISÄÄ DEVOPSISTA

Tiedonjano ei siis päässyt vielä sammumaan. Hyvä! Sukella syvemmälle devopsin maailmaan esimerkiksi näiden tietolähteiden avulla:

THE PHOENIX PROJECT: A NOVEL ABOUT IT, DEVOPS, AND HELPING YOUR BUSINESS WIN (GENE KIM ET AL.) · Kolmen kovan luokan it-ammattilaisen kirjoittama hittiteos, joka kertoo fiktiivisen organisaation selviytymisestä myöhästyneen ja kustannuksiltaan paisuneen projektiin puristuksessa. Vyyhti alkaa purkautua devopsin keinoin...

BUILDING A DEVOPS CULTURE (MANDI WALLS) · Lyhyt ja ytimekäs opas siitä, miten uudenlainen kehittämisen kulttuuri rakennetaan, ja miten muutos saadaan organisaatiossa toteutettua.

DEVOPS.COM · DevOps.com sisältää kattavan kokoelman devops-teemaan liittyviä kirjoituksia aina tietoturvapaatoksesta tiukkaan bisnesasiaan.

EFICODE.FI/BLOGI · Eficoden blogissa myös tämän oppaan kirjoittajat tarjoilevat tuoreita ajatuksia devopsista. Lue vaikkapa kommunikaatiokuilujen ylityksestä tai Suomen kilpailukykyä edistävästä devops-robotista.

EFICODE AUDIT



Yksi tapa aloittaa on Eficoden monessa liemessä testattu auditointipalvelu. Auditoinnissa asiantuntijamme selvittävät puolueettomasti sisäisten tai ulkoistettujen ohjelmistoprojektienne nykytilan ja antavat tarkan kehityssuunnitelman tehostamisesta. Suunnitelmassa käsitellään ohjelmistotuotannon infrastruktuurin, ympäristöjen, prosessien, työkalujen ja kulttuurin solmukohdat. Sen jälkeen voitte päättää toimenpiteistä.

WWW.EFICODE.FI



DEVOPS-KYPSYYSMALLI

WWW.EFICODE.FI

JOHTAMINEN

001
Kehitys on erillään liiketoiminnasta. Uusien kehityshankkeiden aloittaminen on kankeaa.

002
Kehitysprojektit voidaan aloittaa ketterästi ja päätöksentekoon on selkeä strateginen ohjaustapa.

003
Kehitystoimenpiteet voidaan linkittää suoraan yrityksen strategiaan tavoitteisiin. Kokeilevien projektien aloittaminen helppoa.

004
Strategisten digitaalisten tavoitteiden toteutumisesta saadaan reaaliaikaista tietoa.

ORGANISAATIO JA KULTTUURI

Suunnittelu, kehitys ja laadunvarmistus ovat erillään toisistaan. Kommunikointi on pääasiassa kirjallista.

Työ tehdään tiimeissä, mutta kehitys ja laadunvarmistus ovat erillään toisistaan.

Tiimit toimivat itsenäisesti. Heillä on kokonaisvastuu ominaisuuksien kehityksestä ja laadunvarmistuksesta.

Tiimit kommunikoivat säännöllisesti keskenään ja kehittävät työtään yhdessä. Yhteydenpito palveluiden ylläpitoon on jatkuvaa.

YMPÄRISTÖT JA JULKAISU

Tuotteet ovat ympäristökohtaisia ja ne käännetään manuaalisesti. Ympäristöt asennetaan ja konfiguroidaan käsin.

Järjestelmä jakautuu osakokonaisuuksiin ja käännös-ympäristö tunnetaan. Osa julkaisusta on automatisoitu.

Ympäristöjä voidaan asentaa ja konfiguroida automaattisesti. Tuotteen käännös- ja julkaisuprosessit on automatisoitu.

Julkaisu voidaan tehdä automaattisesti ja jatkuvasti. Migraatio- ja palautusprosessit toimivat odotetusti.

BUILD JA JATKUVA INTEGRAATIO

Tuotteen integraatio on automaattista, mutta konfiguraatiota ja käynnistystä hallitaan käsin. Ei artefaktien tai muutoslokien hallintaa.

Prosessi käynnistetään tiimikohtaisesti jokaisesta muutoksesta. Työkalut ovat yhteisiä. Integraatioon ei ole liitetty testausta.

Integraatio kattaa koko tuotteen, ja se on liitetty hyväksymistestaukseen. Riippuvuudet ovat tunnettuja ja hallittuja.

Kehitysorganisaatio tapaa säännöllisesti. Kerätyn metriikan pohjalta pyritään nopeuttamaan palautesykliä ja parantamaan näkyvyyttä.

LAADUNVARMISTUS

Laadunvarmistus tehdään kokonaan käsin ja pääasiallisesti kehitystyön jälkeen.

Yksikkötestaus tai staattinen koodianalyysi on käytössä osassa tuotetta.

Loppukäyttäjille näkyvä toiminnallisuus tai järjestelmä testataan kokonaisuudessaan. Testaajat osallistuvat kehitystyöhön.

Hyväksymistestit esittävät selkokielisesti järjestelmän vaatimukset ja ohjaavat mahdollisimman paljon järjestelmän kehitystyötä.

NÄKYVYYS JA RAPORTOINTI

Raportit tehdään tarvittaessa käsin.

Koodin integraatio, yksikkötestaus ja koodianalyysi ovat näkyvissä tiimille.

Vaatimusten tilaa voi seurata reaaliaikaisesti suhteessa testeihin ja julkaistuihin ominaisuuksiin.

Tuotekehityksessä kerätään metriikkaa, jota käytetään pohjana prosessien parantamisessa.

TEKNOLOGIA JA ARKKITEHTUURI

Teknologia ja työkalut ovat vanhentuneita tai eivät tarkoituksenmukaisia

Teknologia on vanhenemassa ja arkkitehtuuri on vain osin joustava tai rajapinnat ovat puutteelliset.

Teknologia on uudehkoa tai hyvin tuettua. Järjestelmän toiminnot voidaan suorittaa myös rajapintojen kautta.

Arkkitehtuuri ja teknologiavalinnat ovat tarkoituksenmukaisia ja mahdollistavat liiketoiminnallisten tavoitteiden täyttymisen.

EFICODE PIKAOPAS

DEVOPS

Asiantuntijoille

**“Perhaps we think up our own destinies,
and so in a sense deserve whatever happens to us,
for not having had the wit to imagine
something better.”**

Iain Banks (1954 - 2013)

MITÄ ON DEVOPS?

Devops tuo yhteen koko ohjelmistotuotantoketjun aina asiakkaan tarpeista heille toimitettaviin tuotteisiin tai ylläpidettäviin palveluihin saakka.

EFICODE TUTKITUTTI

72 %

Suomalaispäättäjäistä pitää time-to-marketia devopsin tärkeimpänä ominaisuutena.¹

Siinä missä ketterä kehitys keskittyy kommunikaatioon ja ohjelmiston kehittämiseen pienissä kokonaisuuksissa, devops keskittyy asiakkaaseen, automaatioon ja näkyvyyteen. Termi kuvaa tekemisen kulttuuria sekä moderneja työkaluja ja prosesseja.

Devops liittää yhteen kaikki kehitykseen osallistuvat osapuolet: asiakkaan, kehittäjät, ohjelmistotoimittajat, laadunvarmistuksen sekä palveluiden ylläpidon. Monet devopsin periaatteista, kuten jatkuva julkaisu, rikkovat organisaation perinteisiä vastualueita. Devopsissa painotetaan sujuvaa yhteistyötä, jota tyypillisesti tukee keskitetty vaatimustenhallinta ja automaattinen laadunvarmistus.

Devopsin periaatteet eivät ole kokonaan uusia. Monia devopsin osakokonaisuuksia, esimerkiksi virtualisointia ja automaattista laadunvarmistusta, on tehty jo kauan ennen termin keksimistä. Devopsin tarkoituksena onkin tuoda hyvät toimintatavat yhteen ja rakentaa niistä koherentti kokonaisuus.

¹ - Markkinatutkimustoimisto Value Clinic Oy (2014) DevOps Suomessa -tutkimusraportti.



HYÖDYT

MITÄ HYÖTYÄ DEVOPSISTA ON?

Devops yhtenäistää kehityksen ja ylläpidon sekä tuo näkyvyyden asiakkaalle asti:

01. Vaatimukset löytyvät yhdestä selkeästi määritellystä paikasta.
02. Kehitysympäristöt tunnetaan ja ne ovat nopeasti perustettavissa.
03. Tehokas versionhallinta mahdollistaa hallitun kehityksen ja ylläpidon.
04. Toteutettavat ominaisuudet testataan automaattisesti eri tasoilla.
05. Automaattinen laadunvarmistus on yhdistetty alkuperäisiin vaatimuksiin.
06. Tuotteen uusien versio voidaan julkaista täysin automaattisesti tai nappia painamalla.
07. Tuotteen palvelinympäristöt ovat pääosin virtuaalisia. Ne voidaan konfiguroida ja käynnistää automaattisesti.
08. Palveluiden toimintaa ja tehokkuutta voidaan seurata koko organisaatiossa.

ONGELMAT / RATKAISUT

PERINTEISEN
KEHITYKSEN ONGELMA

vs

DEVOPSIN
RATKAISU

Asiakkaalta tulleet vaatimukset, järjestelmän viat ja uudet kehitysjatukset ovat missä sattuu – dokumentaatiosta puhumattakaan.

Järjestelmän integraatio kestää pitkään ja on riskialtista.

Järjestelmän testaus tehdään vasta potentiaalisen versiojulkaisun yhteydessä

Järjestelmän kehittäjät ja tuotantopalvelimien ylläpitäjät eivät tee yhteistyötä.

Palvelinympäristöissä varaudutaan pahimpaan. Tästä syystä kulut nousevat ja joustavuus vähenee.

Kehityssyklit venyvät turhan pitkiksi.

Tuotantoketjun tiimeillä on omat erilliset tavoitteensa, jotka ovat ristiriidassa keskenään.

Tuotantoketjun eri osa-alueet saavat palautetta liian hitaasti ja kohdistamatta.

Jotta yhteistyö on mahdollista, vaatimusten ja dokumentaation pitää löytyä yhdestä paikasta jossa se on kaikille näkyvissä.

Jatkuva integraatio tukee toiminnallisuuden ja yhteensopivuuden varmistusta sitä mukaa, kun tuote muuttuu.

Automaattinen hyväksymistestaus ja jatkuva integraatio mahdollistavat laadunvarmistuksen sitä mukaa kun järjestelmään tehdään muutoksia.

Tuotteen julkaisu on automaattista. Julkaisun toimivuudesta vastaavat kehittäjät ja ylläpito yhdessä.

Palvelinympäristöjen virtualisointi auttaa skaalaamaan palveluita ja hallitsemaan kuluja silloin, kun palvelujen käyttö on vähäisempää.

Automaatio ja tiimien yhteiset tavoitteet varmistavat, että koko tuotantoketju toimii salamannopeasti.

Koko ketjua tarkastellaan yhtenä kokonaisuutena, joka jaetaan pienempiin palasiin.

Automaatiolla kohdistettu palaute tulee hetkessä oikealle henkilölle.

DEVOPSIN KESKEISET PALASET

Devops on yhdistelmä työkaluja ja organisaation yhtenäistä kehityskulttuuria. Devopsin ytimessä on automaatio, jonka avulla tuotekehityksestä voidaan tehdä tasainen virta koko organisaation läpi. Asiakkaan esittämä tarve matkaa automaattisesti työvaiheesta toiseen aina toteutetun ominaisuuden julkaisuun ja ylläpitoon asti.

VAATIMUSTENHALLINTA

Ongelma: Vaatimukset ja bugiraportit ovat hajallaan eri lähteissä.

01. Vaatimukset ja järjestelmän dokumentaatio kerätään yhteen paikkaan koko kehitysorganisaation saataville.
02. Vaatimusten tilaa seurataan koko kehitystyön ajan ohjelmiston julkaisuun asti. Näin tiedetään, milloin ominaisuus on julkaistu ja mitä muutoksia julkaisuun kuuluu.

KEHITYSYMPÄRISTÖT

Ongelma: Kehitysympäristöjen pystyttämiseen menee tolkkottomasti aikaa. Kehittäjillä on myös erilaiset kehitysympäristöt.

01. Kehittäjien ympäristöt tulisi voida asentaa automaattisesti konfiguraatiokirjastoja ja -työkalujen avulla.
02. Kehittäjien ympäristöjen tulisi olla liitetty keskitettyyn vaatimustenhallintaan ja versionhallintaan sekä jatkuvan integraation kautta automaattiseen laadunvarmistukseen ja julkaisuun.
03. Kehitystyötä ohjaa nopea ja automaattinen palaute jatkuvan integroinnin, automaattisen laadunvarmistuksen ja koodianalyysin kautta.

JATKUVA JULKAISU

Ongelma: Tuotantojulkaisujen tekeminen on työlästä, virhealtista ja kestää pitkään.

01. Jatkuva julkaisu pyrkii tilanteeseen, jossa tuote voidaan julkaista tuotantokäyttöön asti jokaisen muutoksen jälkeen automaattisesti.
02. Kehitysvaiheessa käytettyjen konfiguraatioiden tulisi olla yhteydessä palveluiden konfigurointiin ja virtualisointiin, jotta julkaisuprosessi voidaan automatisoida.
03. Jatkuva julkaisu voi olla laatuportteja, jolloin automatisoitu prosessi keskeytyy siihen asti, kunnes tulokset on tarkistettu ja prosessi voi jatkuu.
04. Jatkuva julkaisu on asenne, jossa pyritään automatisoimaan toistuvat prosessit niin luotettavasti, että sen perusteella voidaan julkaista muutoksia palvelun käyttäjille asti.

CASE ROVIO: Vihaiset linnut automatisoidulle ohjelmistotuotannon linjastolle.
eficode.fi/rovio

HYVÄKSYMISTESTAUS

Ongelma: Vaatimuksia ja testejä ei ole yhdistetty. Testaus tehdään pääsääntöisesti käsin ja vasta kehitystyön jälkeen.

01. Vaatimukset yhdistetään laadunvarmistukseen, jotta niiden todellinen tila on koko ajan selvillä.
02. Automatisoidaan ainakin se osa laadunvarmistuksesta, joka pitää tehdä jokaisen muutoksen jälkeen.
03. Kirjoitetaan automaattiset testitapaukset mahdollisimman luonnollisella kielellä, jotta asiakkaat sekä asiantuntijat ymmärtävät, miksi ja miten järjestelmän laatu varmistetaan.

```
Transferring money using a phone number
Login to payment service      ${user}
Input recipients phone number  ${phone}
Input the payment amount      €50
Accept payment
Confirm the payment has been transferred
```

EFICODEN BLOGISSA: Automaatio poistaa pelon hyväksymistestauksesta:
eficode.fi/blogi/hyvaksymistestaus

VIRTUALISOINTI

Ongelma: Palvelinten ylläpito on kallista ja kapasiteettia on reilusti yli normaalin tarpeen. Testi- ja esituotantopalvelinten konfiguraatiot ovat erilaisia kuin tuotantopalvelimilla.

01. Yksi devopsin kulmakivistä on palveluiden virtualisointi.
02. Virtualisointi mahdollistaa ympäristöjen pysymisen identtisenä koko kehityksen ajan.
03. Virtualisoinnilla voidaan pienentää merkittävästi ylläpitokuluja, sillä virtuaalipalvelinten kapasiteettia voidaan skaalata tarpeen mukaan.
04. Virtualisoinnin avulla kehitys ja laadunvarmistus on nopeampaa. Ympäristöistä johtuvia virheitä ilmenee merkittävästi vähemmän.

MONITOROINTI

Ongelma: Palvelu on hidas, mutta kukaan ei tiedä tarkasti miksi.

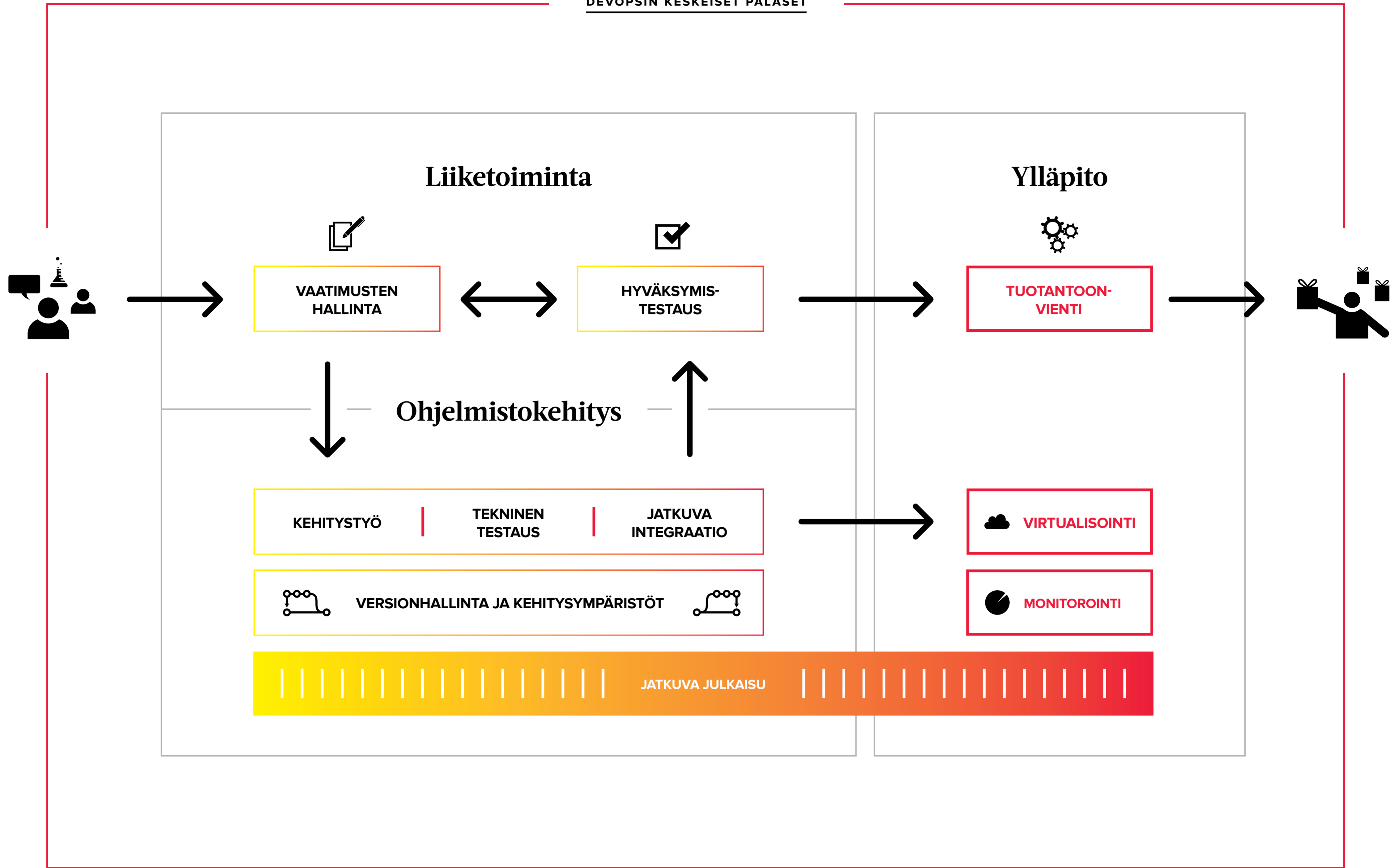
01. Palveluiden monitoroinnilla voidaan seurata sekä palvelimen että itse palvelun suorituskykyä.
02. Nykyaikaiset monitorointityökalut osaavat raportoida suurimmat ongelmakohdat ja avustaa palvelun suorituskyvyn parantamisessa.
03. Monitorointi auttaa kehittämään ja seuraamaan palvelun toimintaa pitkällä aikavälillä.

RAJAPINNAT

Ongelma: Järjestelmää ei voida testata automaattisesti.

01. Devops siirtää tuotekehityksen fokukselta mammuttimaisten järjestelmien toteutuksesta rajapintojen kautta kommunikoiviin pienempiin osakokonaisuuksiin.
02. Rajapinnat dokumentoidaan hyvin ja standardeja noudatetaan.

DEVOPSIN KESKEISET PALASET



KUINKA PÄÄSTÄ ALKUUN?

Voit aloittaa devopsiin tutustumisen vaikka heti. Lähde liikkeelle miettimällä, missä on tuotantoketjusi suurin pullonkaula ja lähde avaamaan sitä jollain edellä mainituista toimintatavoista. Kokeile käsin ajettavien testitapausten automatisointia tai ota selvää voisiko palvelimianne virtualisoida.

HALUATKO TUTUSTUTTAA TIIMISI DEVOPSIIN? KOEILKAA NÄITÄ ASKELIA:

01. Kartoittakaa ohjelmistokehityksenne nykytila totuudenmukaisesti.
02. Rakentakaa roadmap, jota noudattamalla kehitystyötä voidaan tehdä pienissä kokonaisuuksissa.
03. Tehkää kokeiluja alueilla, joiden kulttuuri mahdollistaa muutoksen, tai joissa on eniten hyviä propellipäitä.
04. Tehkää teknisen osaamisen kartoitus ja koeponnistakaa eri menetelmiä.
05. Muodostakaa yhteisiä periaatteita ja tehkää niistä organisaation toimintatapoja.
06. Tehkää hyppy tuntemattomaan! Esimerkiksi: kehittäjät saavat julkaista muutoksia suoraan tuotantoon.

EFICODEN BLOGISSA:

Automatic testing with Robot Framework (video tutorial)

eficode.fi/blogi/automatic-testing

SISÄLLYSLUETTELO



DEVOPS-KYPSYYSMALLI

WWW.EFICODE.FI

	001	002	003	004
JOHTAMINEN	Kehitys on erillään liiketoiminnasta. Uusien kehityshankkeiden aloittaminen on kankeaa.	Kehitysprojektit voidaan aloittaa ketterästi ja päätöksentekoon on selkeä strateginen ohjaustapa.	Kehitystoimenpiteet voidaan linkittää suoraan yrityksen strategiaan tavoitteisiin. Kokeilevien projektien aloittaminen helppoa.	Strategisten digitaalisten tavoitteiden toteutumisesta saadaan reaaliaikaista tietoa.
ORGANISAATIO JA KULTTUURI	Suunnittelu, kehitys ja laadunvarmistus ovat erillään toisistaan. Kommunikointi on pääasiassa kirjallista.	Työ tehdään tiimeissä, mutta kehitys ja laadunvarmistus ovat erillään toisistaan.	Tiimit toimivat itsenäisesti. Heillä on kokonaisvastuu ominaisuuksien kehityksestä ja laadunvarmistuksesta.	Tiimit kommunikoivat säännöllisesti keskenään ja kehittävät työtään yhdessä. Yhteydenpito palveluiden ylläpitoon on jatkuvaa.
YMPÄRISTÖT JA JULKAISU	Tuotteet ovat ympäristökohtaisia ja ne käännetään manuaalisesti. Ympäristöt asennetaan ja konfiguroidaan käsin.	Järjestelmä jakautuu osakokonaisuuksiin ja käännös-ympäristö tunnetaan. Osa julkaisusta on automatisoitu.	Ympäristöjä voidaan asentaa ja konfiguroida automaattisesti. Tuotteen käännös- ja julkaisuprosessit on automatisoitu.	Julkaisu voidaan tehdä automaattisesti ja jatkuvasti. Migraatio- ja palautusprosessit toimivat odotetusti.
BUILD JA JATKUVA INTEGRAATIO	Tuotteen integraatio on automaattista, mutta konfiguraatiota ja käynnistystä hallitaan käsin. Ei artefaktien tai muutoslokien hallintaa.	Prosessi käynnistetään tiimikohtaisesti jokaisesta muutoksesta. Työkalut ovat yhteisiä. Integraatioon ei ole liitetty testausta.	Integraatio kattaa koko tuotteen, ja se on liitetty hyväksymistestaukseen. Riippuvuudet ovat tunnettuja ja hallittuja.	Kehitysorganisaatio tapaa säännöllisesti. Kerätyn metriikan pohjalta pyritään nopeuttamaan palautesykliä ja parantamaan näkyvyyttä.
LAADUNVARMISTUS	Laadunvarmistus tehdään kokonaan käsin ja pääasiassa kehitystyön jälkeen.	Yksikkötestaus tai staattinen koodianalyysi on käytössä osassa tuotetta.	Loppukäyttäjille näkyvä toiminnallisuus tai järjestelmä testataan kokonaisuudessaan. Testaajat osallistuvat kehitystyöhön.	Hyväksymistestit esittävät selkokielisesti järjestelmän vaatimukset ja ohjaavat mahdollisimman paljon järjestelmän kehitystyötä.
NÄKYVYYS JA RAPORTOINTI	Raportit tehdään tarvittaessa käsin.	Koodin integraatio, yksikkötestaus ja koodianalyysi ovat näkyvissä tiimille.	Vaatimusten tilaa voi seurata reaaliaikaisesti suhteessa testeihin ja julkaistuihin ominaisuuksiin.	Tuotekehityksessä kerätään metriikkaa, jota käytetään pohjana prosessien parantamisessa.
TEKNOLOGIA JA ARKKITEHTUURI	Teknologia ja työkalut ovat vanhentuneita tai eivät tarkoituksenmukaisia.	Teknologia on vanhemmassa ja arkkitehtuuri on vain osin joustava tai rajapinnat ovat puutteelliset.	Teknologia on uudehkoa tai hyvin tuettua. Järjestelmän toiminnot voidaan suorittaa myös rajapintojen kautta.	Arkkitehtuuri ja teknologiavalinnat ovat tarkoituksenmukaisia ja mahdollistavat liiketoiminnallisten tavoitteiden täyttymisen.

Liikkeenjohdolle

HEIKKI HÄMÄLÄINEN

Head of DevOps

heikki.hamalainen@eficode.com

+358 (0) 40 766 2610

RISTO VIRKKALA

CEO

risto.virkkala@eficode.com

+358 (0) 40 501 8434

Asiantuntijoille

MARKO KLEMETTI

CTO

marko.klemetti@eficode.com

+358 (0) 44 522 5927

KAJ JOKINIEMI

Vice President of DevOps

kai.jokiniemi@eficode.com

+358 (0) 40 592 6257



Eficode Oy

Pohjoinen Rautatiekatu 25, 00100 Helsinki Finland - +358 (0) 207 40 11 22

Eficode on Helsingissä, Pekingissä ja Kööpenhaminassa toimiva ohjelmistotalo, jonka toimintaa ohjaa yksi periaate yli muiden: ei hölynpölyä vaan tuloksia.