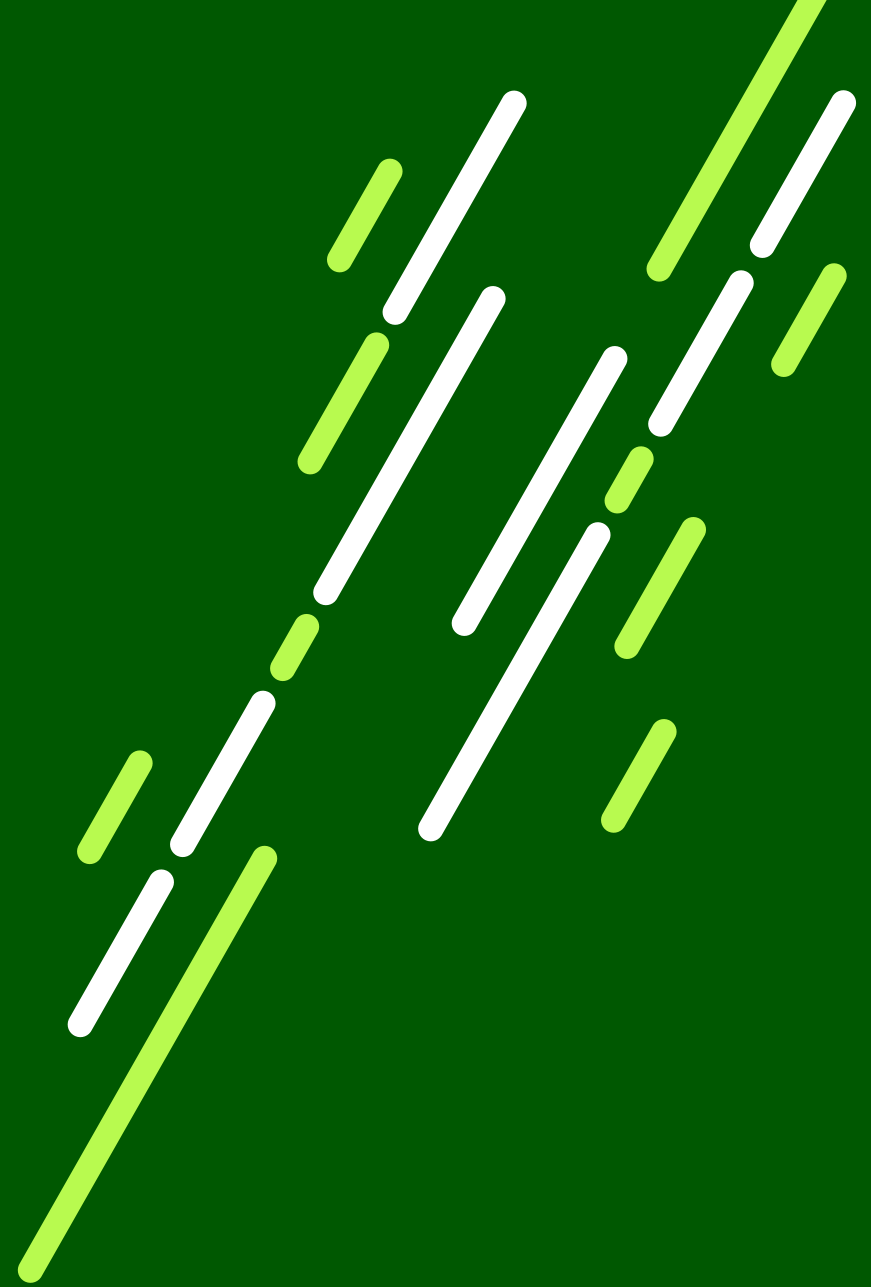


**netum**

# MitViDi -hankkeen arviointikehikön ja mittausjärjestelyn selvityksen esittely

Aki Nordlund & Mikko Satama 20.9.2023



# Esittelyt

## Aki Nordlund

Management Consultant



- Monipuolinen ja laaja yli 25 vuoden kokemus liiketoiminnoista ja liiketoimintamalleista, johtotehtävistä, arvoverkostoista sekä ekosysteemeistä ICT-alalta.
- Kaksoissiirtymä, vastuullisuus ja kestävät liiketoiminnot lähellä sydäntä.

## Mikko Satama

Yritysarkkitehti



- Yli 25 vuoden monipuolinen työkokemus ICT-alalta eri roolien kautta, joissa painopiste on enimmäkseen ollut tuote- ja sovelluskehityksessä.
- Palava into edistää asioita ketterästi ja viisaasti maalaisjärkeä hyödyntäen, jolloin lopputuotokset ovat tarkoituksenmukaisia, uudelleenkäytettäviä sekä resurssitehokkaita.

# Netum – Viisas muutoksen tekijä

Kannattavasti kasvava IT-palvelutalo yli 20 vuoden kokemuksella vaativista IT-hankkeista

Netum on luotettava digipalvelukumppani, joka tarjoaa asiakkailleen sopivimmat ratkaisut ja sujuvimman arjen.

Autamme asiakkaitamme digitaalisen liiketoiminnan kehittämisessä suunnittelemalla ja toteuttamalla juuri heidän tarpeisiinsa sopivia ja kestäviä digiratkaisuja

– viisaasti, vastuullisesti ja turvallisesti.

TOIMINTA ALOITETTU

2000

LIKEVAIHTO 2022

29,1 M€

TYÖNTEKIJÖITÄ (09-2023)

430+

NPS

64

LISTATTU NASDAQ  
HELSINGISSÄ (First North)

2021



# Asiakkaitamme ja kumppaneitamme



# Netum Sustainability

## Vastuullista digimuutosta

- Tavoitteenamme on olla alan luotetuin kumppani ja auttaa asiakkaitamme digitaalisessa muutoksessa viisaasti, vastuullisesti ja turvallisesti.
- Roolimme toimijana on olla moderni ja skaalautuva palvelun tarjoaja.
- Toimintafilosofiaamme kuuluu uusien digitaalisten palvelujen kehittäminen siten, että yhdistämme käytössä olevat perinteiset tietotekniikkaratkaisut uusimpiin digitaalisiin sovelluksiin.
- Vihreät toimintamallit ovat meille strateginen valinta. Haluamme tulla tunnetuksi digitaalisten palveluiden vihreänä kehittäjänä ja edelläkävijänä Suomessa.
- Vihreän koodaamisen ja suunnittelun tavoitteena on rakentaa palveluja ja tuotteita vastuullisin menetelmin.

# Lähtökohdat selvitykselle



- MitVidi-hankkeen tavoitteena on luoda mittareita kestävämpien ICT- hankintojen tukemiseen. Hankkeen painopiste tässä arvioinnissa on julkisissa hankinnoissa.
- Tässä selvityksessä keskitytään laadullisten ominaisuuksien edistyneen tason reflektointiin sekä tason järjestelmäarkkitehtuuriin arvioimalla pystyvätkö toimittajat vastaamaan arviointikehikon kriteereihin.
- Selvityksessä arvioidaan lisäksi, onko kuvauksen perusteella vastaavan tyyppisiä mittauksia toteuttaa sekä arvioida mittauksien järkevyyttä.



- Kohdejärjestelmänä Peppi
- Korkeakoulujen (YO & AMK) toiminnanohjausjärjestelmä opintojen ja opetuksen hallintaan
- Roolipohjainen käyttöliittymä (erilliset työpöytänäkymät opiskelijoille, opettajille, suunnittelijoille ja ylläpitäjille)
- Palvelukeskeinen Java-teknologialla toteutettu arkkitehtuuri
- Netum mukana käyttöönotoissa, ylläpidossa (sovellus ja infra) sekä sovelluskehityksessä
- <https://www.peppi-konsortio.fi/>

# Arvointikehikko

Mittarikortit

Edistynyt taso: Laadulliset arkkitehtuuritason  
ominaisuudet

Pystyvätkö toimittajat näihin vastaamaan?



# 2.1 Toimivat rajapinnat

Ohjelmiston rajapinnat ovat käytettäviä ja jatkuvia, ohjelmiston API on selkeästi dokumentoitu

## Kohdejärjestelmän näkökulmasta

- ✓ Ohjelmiston palvelumoduulit on toteutettu REST- ja SOAP-rajapinnoilla ja julkaistu OSGi-rekisteriin
- ✓ Kehittäjille suunnattu dokumentaatio löytyy
- ✓ Käyttöliittymäkerros toteutettu Liferay-julkaisujärjestelmästä, jonka dokumentaatio kunnossa
- ✓ 3. osapuolien on mahdollista tehdä jatkokehitystä
- × Onko dokumentaatio riittävän selkeää ulkopuolisille ja tarvitseeko sen olla.
- × Java-riippuvuus



## Huomiot ja havainnot vaatimuksesta

- ✓ Nykypäivänä pakollinen vaatimus
- ✓ Varsinainen mittarikortti avaa vaatimuksen hyvin
- × *"käytettäviä ja jatkuvia"*, *"selkeitä"* – saadaanko näitä mitattavimmiksi, jolloin vertailu helpottuisi
- × Mikäli rajapinta on hyvin dokumentoitu, mutta toteutustapa ei-standardi → hyvä vai huono?
- × Kumpi on vihreämpi: helposti laajennettava massiivinen palvelu vai tehokkaasti määrättyä asiaa hoitava ketterä palvelu, joista koostamalla saa laajemman toiminnallisuuden.





# 2.2 Ohjelmiston taaksepäin yhteensopivuus

Ohjelmisto on suoritettavissa vanhalla referenssijärjestelmällä

## Kohdejärjestelmän näkökulmasta

- ✓ Yleensä minor-päivitykset pystytään ajamaan samassa palvelinympäristössä
- ✓ Referenssijärjestelmää ei määritelty
- ✓ Uusien toiminnallisuuksien lisääminen on yleensä tarkoittanut myös kasvavaa resurssitarvetta
- ✓ Liferay-alusta asettaa omat vaatimuksensa käyttöympäristölle
- ✓ "...resurssisyöppö ja optimoinnille on tarvetta"

## Huomiot ja havainnot vaatimuksesta

- ✓ Tarkoituksena hyvä eli pidentää laitteistojen käyttöikä
- ✓ Toimii, mikäli ohjelmistoa ajetaan perinteisessä palvelinympäristössä
- ✓ Toimiiko, mikäli ohjelmistoa ajetaan pilvitoimittajien IaaS tai PaaS –palveluiden päällä
- ✓ Vasta itse mittarikortista selviää, että referenssijärjestelmällä tarkoitetaan laitteistoja (ja varusohjelmistoja ja käyttöjärjestelmiä) ☒ otsikointia voisi ehkä muuttaa



# 2.3 Tulevaisuuden ennakointi

Ohjelmiston data ja dataformaatit siirrettävissä toisiin ohjelmistoihin, ohjelmiston jatkuvuus ja pitkä elinikä

## Kohdejärjestelmän näkökulmasta

- ✓ Palvelun rajapinnat ja toteutus on eriytetty toisistaan
- ✓ Jokaisella palvelulla on oma SQL schema
- ✓ Palveluiden tekemät SQL-kyselyt on suunniteltu tietokantariippumattomiksi
- ✓ Tietoja käsitellään vain rajapintojen kautta
  
- ✓ Tietokantayhteyksiin käytetään Spring JDBC -tekniikkaa

## Huomiot ja havainnot vaatimuksesta

- ✓ Kun puhutaan vain tietovaraston siirrettävyydestä, vaatimus kuulosta järkevältä – mikäli tällä tarkoitetaan migraatiota esim. toiseen tietokantateknologiaan.
  
- ✓ Pitäisikö ”integroitavuudesta” muodostaa kokonaan oma vaatimuksensa helpommin arvioitavammalla kriteeristöllä (HUOM! Vahva linkitys 2.1 Toimivat rajapinnat)



# 2.4 Avoimuus

Läpinäkyvyys ja selkeys ohjelmiston lähdekoodissa, ohjelmistossa ja sen komponenteissa, ohjelmiston dokumentaatiossa ja käyttöehdoissa

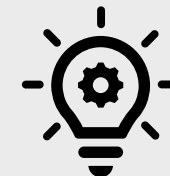
## Kohdejärjestelmän näkökulmasta

- ✓ Pohjautuu avoimeen lähdekoodiin ja sovelluskomponentteihin
- ✓ Arkkitehtuuria evaluoidaan muihin saman toimialan toteutusprojekteihin
- ✓ Kehittämiskäytäntöjä ylläpidetään
  
- ✓ Dokumentaation selkeys – voi olla vaikea hahmottaa kokonaisuutta



## Huomiot ja havainnot vaatimuksesta

- ✓ Mikäli läpinäkyvyydellä tarkoitetaan ohjelmiston tiekarttaa, vaatimus siltä osin selkeä.
- ✓ Tällä vaatimuksella voidaan taklata myös nykypäivälle hyvin tyypilliset "komponenttiketjutukset" eli komponenttien väliset riippuvuudet.
  
- ✓ Tästä ei löytynyt vielä tarkempaa mittarikorttia
- ✓ "läpinäkyvyys" ja "selkeys" voivat olla vaikeasti vertailtavia
- ✓ Huomioitava myös muu kuin ohjelmistokoodin dokumentaatio (kokonaisarkkitehtuuri)



# 2.5 Optimointi

Tuote sisältää tiedot, joilla vähentää sen resurssienkulutusta, ohjelmiston koodi ja muu toteutus on optimoitu resurssienkulutuksen minimoimiseksi, ohjelmiston toiminnot perustuvat käyttäjien todellisiin tarpeisiin.

## Kohdejärjestelmän näkökulmasta

- ✓ -
- ✓ Sovellusvalvontaa (käytettävyys ja suorituskyky) ei tällä hetkellä ole käytössä



## Huomiot ja havainnot vaatimuksesta

- ✓ Erittäin tärkeä vaatimus!
- ✓ Loppukäyttäjien todellisten tarpeiden tunnistaminen tärkeää jo ennen tarjouspyyntövaihetta (tilaajan vastuu)
- ✓ Sovellusvalvonnasta saadaan tärkeää lähtötietoa kehitystoimenpiteille sekä pystytään arvioimaan muutosten vaikutusta
  
- ✓ Yksittäinen käyttäjätesti ei ehkä ole riittävä vaan tarvitaan jatkuvaa seuranta ja valvontaa



# Yhteenveto arvointikehikosta/reflektoinnista

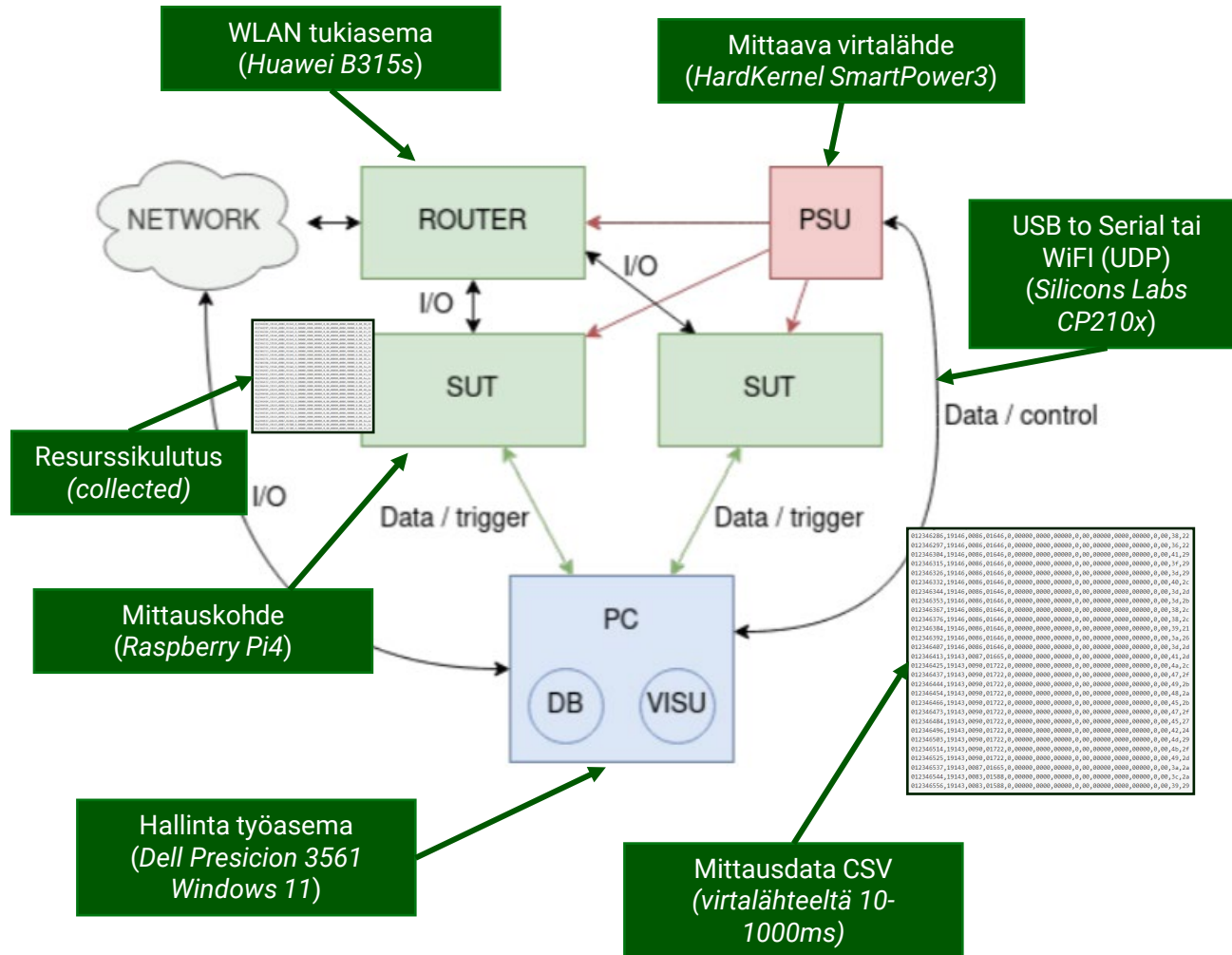
- Helpottaa tarjousvertailua
  - Pakottaa toimittajat ajattelemaan vihreästi
  - Vaatimukset tarkennettava riittävälle tasolle
  - Omat vaatimukset erityyppisille hankkeille?
- 
- Heti käyttöön!

# Mittausjärjestely - Matkalla huipputasolle

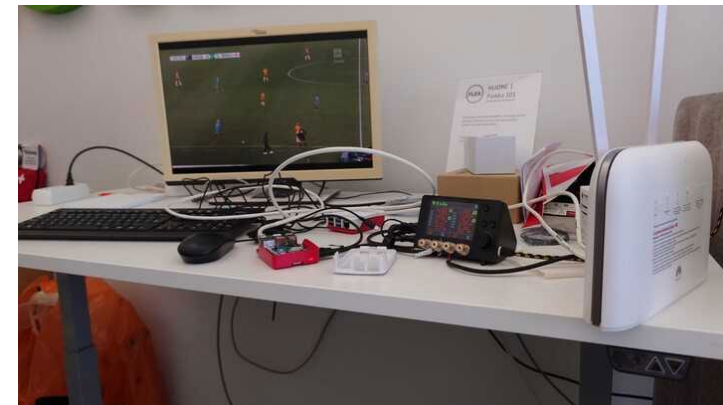
Toteutuskelpoisuus?  
Onko järkeä vai ei?



# Mittausjärjestely – kokoonpano ja prosessi



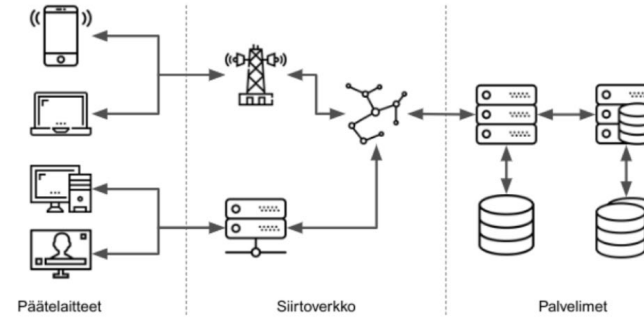
- SUT-laitteiden (Raspberry) virrankulutustiedot kerätään PSU (SmartPower3) laitteella
- SUT-laitteiden resurssikulutustiedot (prosessori- ja muistikuormitus) kerätään talteen
- Kerätyt tiedot analysoidaan ja raportoidaan työasemalla



# Toteutus ja havainnot

Ei mennyt suunnitelmien mukaan

1. Laitteiston hankinta
2. Laboratorion pystytys
3. Virtamittarin toiminnan opettelu
4. Kohdejärjestelmän valinta
5. Mittausdatan kerääminen
  - "Linux, linux, linux ja linux..."
  - "Entäs jos kuitenkin Windows"
  - "Linux, linux, linux ja linux..."
  - ...
6. Henkinen luovuttaminen
7. Back-end virrankulutuksen mallintaminen



- ✓ Mittausjärjestely voisi olla toimiva, jos
  - Vakioitu laitteisto + linux image
  - Datan keruuscriptit valmiina (sis. kohdistuksen)
  - Myös back-end ja verkkoliikenne mukana
  - Raportointi näkymät valmiina
- ✓ Raspberry-laitteiston alhainen virrankulutus (~2-4W vs. Windows työasema ~10-100W)
- ✓ Toimii varmasti laitteiston virranmittauksessa
- ✓ Saadaanko yksittäinen käyttäjän toiminto selaimessa kohdistettua yksittäiseen virtapiikkiin





# netum

Vaihde: 041 529 5555 | Myynti: myynti@netum.fi | Sähköpostiosoitteet: etunimi.sukunimi@netum.fi | [www.netum.fi](http://www.netum.fi)

Helsinki | Tampere | Turku | Pori | Jyväskylä | Hämeenlinna | Kuopio | Oulu