

Opinnäytetyö (AMK)

Elektroniikka

2010

Risto Kontiokoski

MAINTENANCE MONITORING TOOL

- ohjelmisto- ja käyttöönottoprojekti



TURUN AMMATTIKORKEAKOULU
TURKU UNIVERSITY OF APPLIED SCIENCES

OPINNÄYTETYÖ (AMK) | TIIVISTELMÄ

TURUN AMMATTIKORKEAKOULU

Elektroniikka

Kesäkuu 2010 | Sivumäärä 33

Ohjaaja | Yngvar Wikström, Pasi Pekkanen

Risto Kontiokoski

MAINTENANCE MONITORING TOOL

- ohjelmisto- ja käyttöönottoprojekti

Tässä työssä otettiin käyttöön uusi ohjelmisto Maintenance Monitoring Tool tuotannon kunnossapito-osastolle sekä kehitettiin ohjelmistoa toimivammaksi ja tehokkaammaksi loppukäyttäjän näkökulmasta. Hankkeeseen kuului järjestelmän käyttöönotto, huoltorajojen määrittely sekä ohjelmiston jatkokehitys ja ohjeistukset. Työkalun avulla hallitaan aika- sekä käyttökertoihin perustuvia huoltoja, analysoidaan kertyvää dataa sekä kehitetään huoltojärjestelmää

Projekti toteutettiin viiden etapin avulla. Ensimmäisessä vaiheessa määriteltiin projektin sisältö, vastuut ja aikataulus. Projektin toisessa vaiheessa ohjelmisto otettiin käyttöön yhden tuotteen osalta. Kyse oli tällöin ohjelmiston toiminnallisuustestauksesta. Kolmannessa vaiheessa ohjelmisto otettiin käyttöön kaikkien tuotteiden osalta, luotiin toimintamalli sekä tehtiin ohjeistukset työkalulle. Neljännessä vaiheessa kertynyttä dataa analysoitiin sekä aikaperusteisia huoltoja siirrettiin Maintenance Monitoring Tool -työkaluun muuttamalla ne kappalemääräperusteisiksi. Viimeisessä vaiheessa dokumentit ja käyttöohjeet päivitettiin, sekä raportoitin loput kehitysideat ohjelmistonkehitysosastolle. Projekti toteutettiin täysin aikataulussa ja saatiin käyttöön koko ennalta määrätyle alueelle.

Lopputuloksena syntyi hyvä implementointimalli, joka sisältää menetelmän käyttöönottoa ja kehitystä tukevat tekijät. Projektin aikana syntyi suuri määrä parannusideoita, joilla ohjelmasta saadaan vieläkin tehokkaampi. Mallia voidaan hyödyntää yleisemmin tuotannossa, silloin kun on kyse uuden menetelmän käyttöönotosta tai ohjelmiston jatkotutkimuksesta.

ASIASANAT: kunnossapito, ennakkohuollot, projekti

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Electronics

June 2010 | Total number of pages 33

Instructor: Yngvar Wikström, Pasi Pekkanen

Risto Kontiokoski

A MAINTENANCE MONITORING TOOL

-A software and implementation project

The purpose of this thesis was to implement a new software Maintenance Monitoring Tool and to develop software that works better and more efficiently from the end-users's perspective. The purpose of this tool is to manage time based maintenance as well as to separate usage, analyze accumulated data and to develop the maintenance system. The project included system installation, defining maintenance limits to parts as well as tool development and guidance.

The project was realized in five steps. In the first phase project content, responsibilities and scheduling were defined. In the second phase the software was implemented for one product. It was a case of software functionality testing. In the third phase the software was implemented for all products, a process model was created and the guidance for the tool was given. The fourth phase was about analyzing the accumulated data and transferring time-based maintenances to the Maintenance Monitoring Tool. At the Final stage of the project documents and instructions were updated, and the rest of the developing ideas were reported to Software Development Department.

The result of this thesis was a implementation model, which includes the factors supporting the implementation and development of the process. The carefully maintained maintenance system enables the efficient and secure operation of the testing process at the company. The model can be used generally in production, in the case of implementing a new method or a software follow-up study.

KEYWORDS: maintenance, project

SISÄLTÖ

1 JOHDANTO	1
2 TESTAUSPROSESSI	2
2.1 Panel Flash -työvaihe	3
2.2 Flash & Alignment -työvaihe	3
2.3 Final & User Interface -työvaihe	5
2.4 Label-työvaihe	6
3 KUNNOSSAPITO	7
3.1 Ennakoiva kunnossapito	8
3.2 Korjaava kunnossapito	8
3.3 Valmistajan käyttörajat	8
3.4 Parantava kunnossapito	9
4 KÄYTTÖÖNOTTOPROJEKTIN SUUNNITTELU	10
4.1 Projektiehdotus	10
4.2 Projektisuunnitelma	10
4.3 Etappikatselmoinnit	11
4.4 Ajan käyttö	12
4.5 Riskikartoitus	13
4.6 Investoinnit	14
5 MAINTENANCE MONITORING TOOL KÄYTTÖÖNOTTOVAIHE	15
5.1 MMT-toimintaperiaate	16
5.2 Projektin alustus	18
5.3 Projektin tavoitteet	18
5.4 Yrityksen vaatimukset ohjelmistolta	19
5.5 Ohjelmiston testaus	19
5.6 Tuotteiden konfigurointi	19
5.7 Tilastollisen aineiston hyödyntäminen ja ennakkohuoltorajat	20
5.8 Käyttöopastus	22
5.9 Aikaperusteisten huoltojen siirtäminen työkaluun	23
6 KEHITYSIDEAT JA PALAUTE	24
6.1 Palaute	24
6.2 Varaosanumerot	24
6.3 Adapterien muistit	24
6.4 Raportointiliittymän hitaus ja online tila	25

6.5 Vuorontiedon näkyminen työkalussa	25
6.6 Laiteluettelonäkymä	25
7 ARVIOINTI JA POHDINTA	27
7.1 Tulokset	27
7.2 Työn ja menetelmien arviointi	27
7.3 Työn tulosten luotettavuus ja jatkojalostus	28
7.3.1 Väriytyksen hyödyntäminen käyttöliittymässä	28
7.3.2 Raportit	29
7.3.3 Raportointisovelluksen historiatiedot	29
7.3.4 Laskuri linjan sekä solun juureen	29
7.3.5 Ohjelman vastuhenkilö	30
7.3.6 Käyttöseuranta	30
7.3.7 Testauslaitteiden kalibroinnit	30
7.3.8 Dokumentit	31
8 YHTEENVETO	32
LÄHTEET	33

Sanasto

adapteri	Mekaaninen rajapinta, joka toimii testattavan kaitteen ja järjestelmän välillä
ATO	Assembly To Order.
FinUI	Testausvaihe, jossa valmis engine testataan ennen ATO:on lähetystä. Ks. Engine. (Final & User Interface)
fixture	Testauslaite, johon adapteri asetetaan. Ks. Adapteri.
flali	Testausvaihe, jossa puhelimen moduuli viritetään ja testataan. (Flash & Alignment)
flash	Haihtumaton muisti
FPS	Flash-ohjelmointilaite (Flash Programming System)
IMEI	International Mobile Equipment Identification
label	Työvaihe, jossa puhelimeen laitetaan päällikuoret ja asennetaan maakohtaiset asetukset.
LATO	Tuotannon alue, jossa puhelimet viimeistellään, pakataan ja lähetetään ostajalle (Lean Assembly To Order)
LENO	Tuotannon alue, jossa valmistetaan engineitä (Lean Engine Operations).
MFR	Tuotannon vikaprosentti (Manufacture Failure Rate)
Panel Flash	Testausvaihe, jossa piirilevypaneelissa olevat flashmuistipiirit ohjelmoidaan.
PDMS	Tietokanta, johon kerätään tietoa puhelimen koko tuotantoprosessin ajalta (Production Data Management System).
PowerMaint	Tietokantapohjainen kunnossapito- ja materiaalinhallintaohjelma.
PSN	Tuotannon sarjanumero (Production Serial Number)
RF	Radiotaajuus (Radio Frequency)
SMT	Pintaliitoskomponenttien ladonta ja juottaminen (Surface Mount Technology)
WLAN	Wireless Local Area Network, langaton lähiverkko jolla erilaiset päätelaitteet voidaan yhdistää ilman kaapeleita

1 JOHDANTO

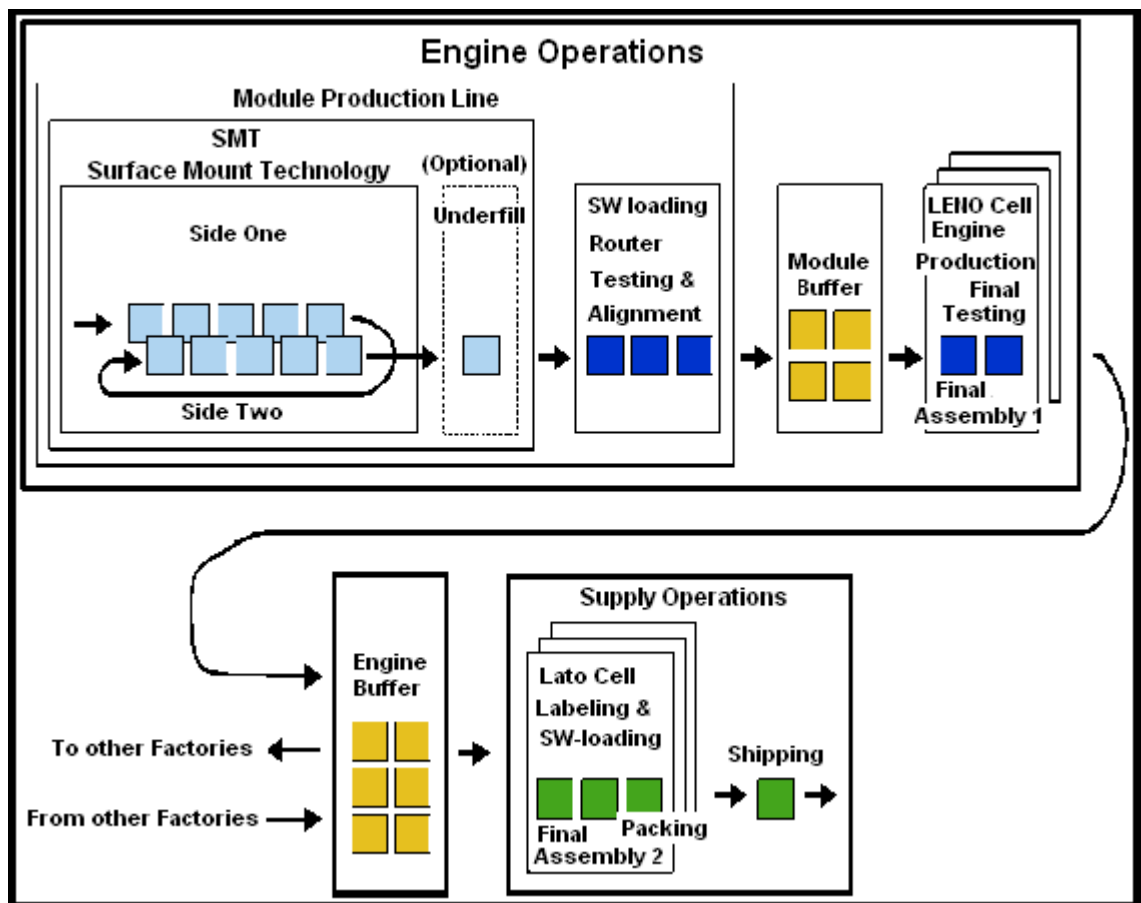
Tuotannossa matkapuhelimen testauksen tarkoituksena on varmistaa, että matkapuhelin täyttää sille asetetut vaatimukset. Käytännössä tämä tarkoittaa erilaisilla testilaitteilla suoritettavia testejä ja virityksiä, joilla havaitaan prosessissa vikaantuneet sekä materiaaliperäiset virheet. Virityksillä puhelin saadaan vastaamaan määriteltyjä vaatimuksia. Testauksen on tapahduttava mahdollisimman nopeasti ja kustannustehokkaasti. Tehtaalla on käytössä yhtenäinen testauslaitteisto, jota muutetaan matkapuhelinmallin mukaan vastaamaan puhelimelle määriteltyjä testausvaatimuksia.

Aiemmin yrityksellä on ollut käytössä erilaisia huoltolaskureita, joiden avulla ennakkohuollot on ajoitettu. Laskureiden ylläpitäminen on ollut työlästä eikä niissä ole ollut kunnollisia tiedon analysointityökaluja. Yrityksellä käytössä olleet huoltolaskurit ovat syntyneet lähinnä opinnäytetöinä. [2] [3] [10] Yritys päätyi teettämään kokonaan uuden ohjelmiston joka vastaisi paremmin huolto-osaston tarpeita. Aiemmin testilaitteet syötettiin manuaalisesti ja jokainen paikkamuutos täytyi konfiguroida erikseen järjestelmään. Tämän vuoksi laskurit olivat usein päivittämättä ja laskureiden käyttö oli vähäistä. Uusi työkalu päivittää paikkatiedot oikeiksi ja siirtää huoltotiedot automaattisesti testilaitteen mukana kunhan ne on konfiguroitu kerran järjestelmään. Lisäksi uusi järjestelmä sisältää monipuoliset tiedonanalysointityökalut, aikaperusteiset huollot ja hälytysviestien lähettämismahdollisuuden huollon lähestyessä, joita ei aiemmissa järjestelmissä ole ollut.

Ohjelmiston varsinainen kehitys alkoi noin 1,5 vuotta ennen projektin aloittamista ja halutut ominaisuudet oli määritelty yrityksen asiantuntijoiden toimesta. Työkalu tulee jatkossa käyttöön myös yrityksen muissa toimipisteissä eri maissa. Työssä keskityttiin matkapuhelimen valmistuksen alkupään testausalueen käyttöönottoon, jonne työkalu ensimmäisenä otettiin käyttöön. Työ suoritettiin projektiluontoisena kehittämistyönä.

2 TESTAUSPROSESSI

Yrityksessä testausprosessi jakaantuu kahteen osa-alueeseen: LENO (Lean Engine Operations) ja LATO (Lean Assembly to Order). Lean-ajattelu on johtamisfilosofia, joka keskittyy seitsemän hukan poistamiseen tuotannossa. Näihin turhuuksiin lasketaan kuljetukset, varastot, liike, odotusaika, ylituotanto, yliprosessointi ja viallinen tuote. LENO-vaiheessa valmistetaan matkapuhelimen runko eli matkapuhelin ilman päällimmäisiä kuoria ja tilaajakohtaista ohjelmistoa. LATO-vaiheessa tuote kootaan sekä puhelimeen ohjelmoidaan varioivia ohjelmapaketteja tilaajalle sopivaksi kokonaisuudeksi. Tässä työvaiheessa matkapuhelin myös pakataan lopulliseen myyntipakkaukseensa. Kuvassa 1 on kuvattu matkapuhelimen tuotantoprosessi se miten eri vaiheet sijoittuvat siihen.



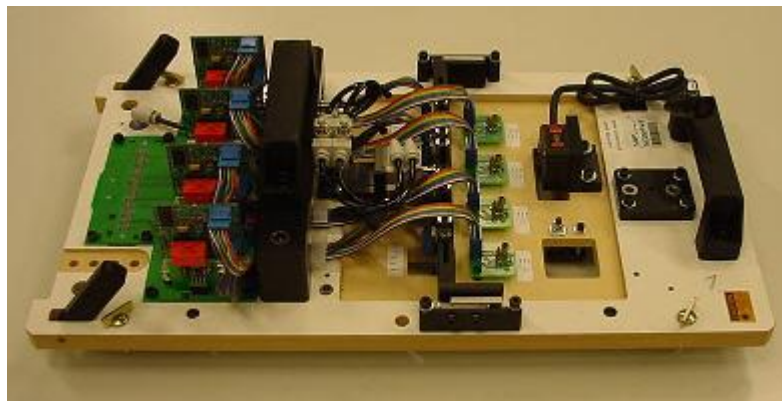
Kuva 1. Tuotantoprosessi ja työvaiheet.

Käytännössä edellä mainitut vaiheet jakaantuvat vielä pienemmiksi osa-alueiksi, joita testauksen näkökulmasta katsottaessa ovat ohjelmointi (Panel Flash), jossa piirilevypaneelissa olevat flashmuistipiirit ohjelmoidaan. Esitestaus (Flash & Alignment),

jossa puhelimen moduuli viritetään ja testataan. Lopputestaus (Final & User Interface), jossa valmis matkapuhelimen runko testataan ennen LATO:on lähetystä. Pakkaus ja loppuohjelmointi (Label), jossa puhelimeen laitetaan päällikuoret ja asennetaan maakohtaiset asetukset).

2.1 Panel Flash -työvaihe

Piirilevy saapuu koneladonnasta Panel Flash -työvaiheeseen, joka on siis testauksen ensimmäinen vaihe. Panel Flash -työvaiheessa piirilevyjen muistipiireille ohjelmoidaan matkapuhelimen perusohjelmisto, joka ohjaa loppukäyttäjällä matkapuhelimen perustoimintoja. Perusohjelmisto aktivoi matkapuhelimen piirit ja tämän vaiheen jälkeen puhelin voidaan ensimmäisen kerran käynnistää. Yhdellä ohjelmitavalla piirilevyaihiolla on 3 – 4 matkapuhelinta. Kaikki aihion matkapuhelimet ohjelmoidaan saman aikaisesti. Panel Flash -työvaiheessa puhelin saa tuotannon sarjanumeron, jonka avulla puhelinta voidaan seurata kaikissa tuotantovaiheissa. Panel Flash -työvaiheesta piirilevykortti siirtyy jyrsimelle, joka erottelee piirilevyaihion yksittäisiksi matkapuhelimiksi. Tämän jälkeen yksittäiset matkapuhelimet siirtyvät Flali-työvaiheeseen. Panel Flash -adapterissa (kuva 2) on ennakkokuullon piiriin kuuluvina osina ainoastaan testauspiikit, jotka on vaihdettava, kun ne ovat kuluneita. [1]



Kuva 2. Panel Flash -adapteri. [2]

2.2 Flash & Alignment -työvaihe

Flash & Alignment -työvaiheessa eli Flali-työvaiheessa ja tästä eteenpäin puhelimia käsitellään yksittäin. Flali-työvaiheessa puhelin viritetään vastaamaan vaadittuja standardeja ja rajoja. Flali-työvaiheessa suoritetaan myös erilaisia testauksia, jolla

varmistetaan että puhelin on valmis jatkamaan kokoonpanoon ja siitä edelleen seuraavaan työvaiheeseen. Flash-muisti on myös mahdollista ohjelmoida Flali-työvaiheessa ja silloin Panel flash -työvaihe jää pois. Työvaiheen jokaisessa testilaitteessa on joko yksi tai kaksi adapteria riippuen tuotteesta. Kuvissa 3 ja 4 näkyvät Flali-adapterit ovat tuotekohtaisia ja sisältävät erilaisen määrän ennakkohuollon piiriin kuuluvia osia tuotteen mukaan. [1]



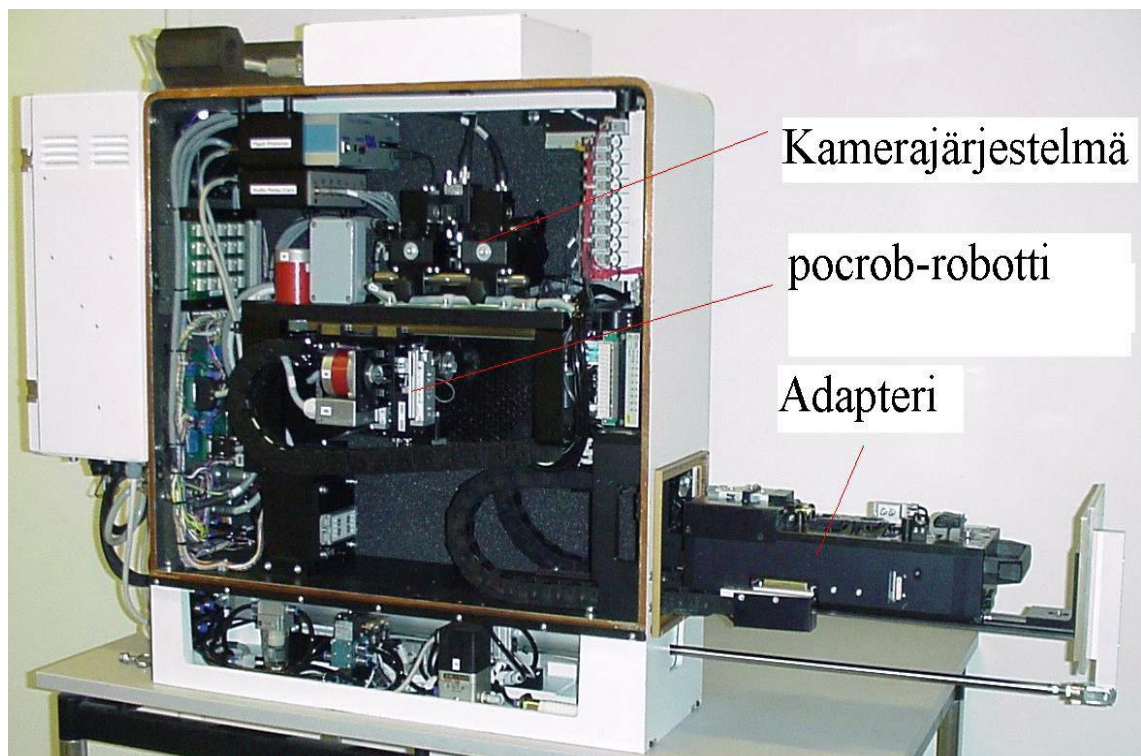
Kuva 3. Flali-testausvaiheen ala-adapteri. [2]



Kuva 4. Flali-testausvaiheen yläadapteri. [2]

2.3 Final & User Interface -työvaihe

Final & User Interface -työvaiheessa eli FinUI-työvaiheessa matkapuhelin on jo koottuna ilman näppäimistöä, kuoria ja akkua. Puhelinta kutsutaan tässä kokoonpano- ja testausvaiheessa Engineksi. FinUI-työvaiheessa mittalaitte varmistaa, että Flali-työvaiheessa viritetyt testiarvot ovat kohdallaan ja tuote on läpäissyt edellisen testivaiheen. Lisäksi UI-testissä (User Interface=käyttöliittymä) testataan käyttäjäkohtaiset lisätoiminnot, kuten esim. lähiverkon, Bluetoothin, näppäimistön, kuulokkeen, kameran ja näytön toimivuus. Kuvassa 3 on Finui-testimekaniikka ja adapteri. [1]

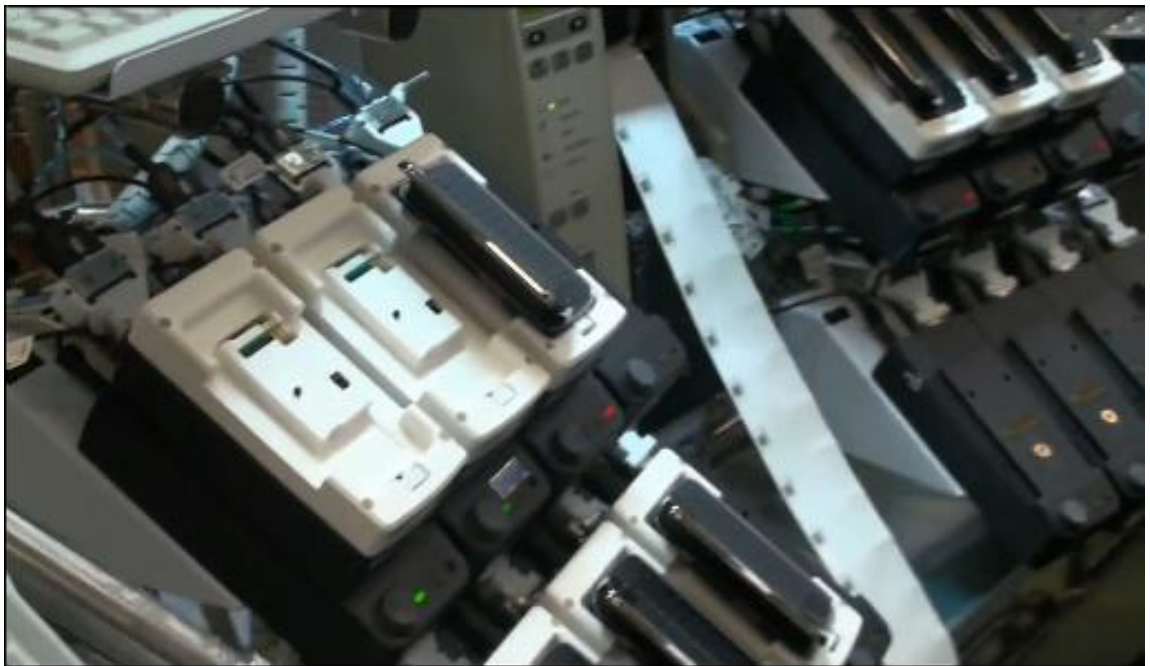


Kuva 5. Finui-testauslaitteisto. [3]

Finui-testauslaite sisältää yhden adapterin ja mekaniikan. Tässäkin vaiheessa adapterit ovat tuotekohtaisia ja vaihtuvat matkapuhelinmallien mukana. Testilaitteeseen kuuluu lisäksi testiräkki johon on sijoitettu mittalaitteet. Testimekaniikat pysyvät samoina jokaisella tuotteella.

2.4 Label -työvaihe

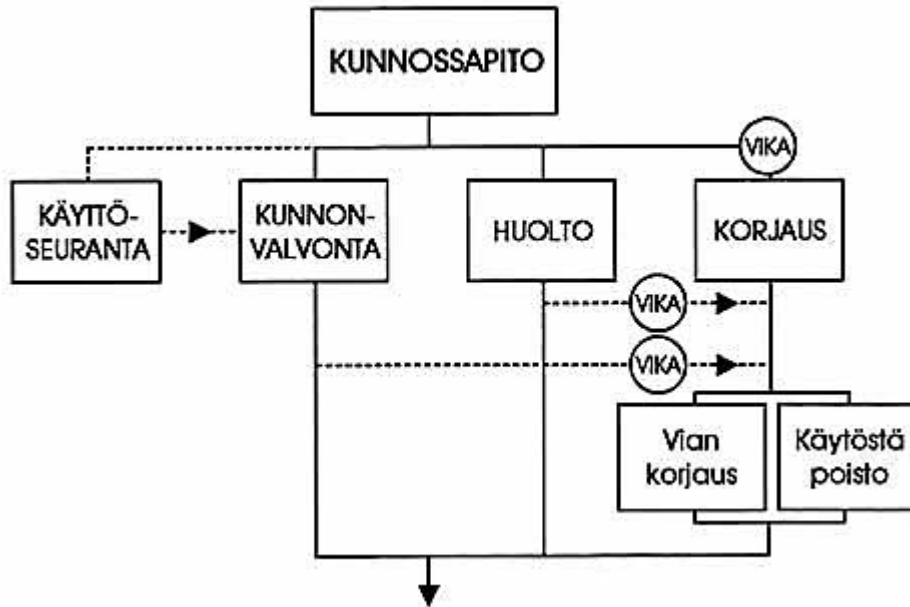
Label on puhelimen viimeinen työvaihe. Tässä testausvaiheessa engine saa lopulliset ulkokuoret, näppäimistön sekä tilauskohtaisen ohjelmistopakettin. Varioitavia ovat esimerkiksi kielipaketit, soittoäännet ja kuvat. Puhelin saa myös IMEI-koodinsa (International Mobile Equipment Identification), joka on kansainvälisesti käytössä oleva sarjanumero mobiililaitteen tunnistamiseksi. Tämän jälkeen matkapuhelin pakataan myyntipakkaukseen oheislaitteineen joita ovat esimerkiksi akku, laturi ja ohjekirja. Kuvassa 6 on Label-työvaihe. [1]



Kuva 6. Label-työvaihe.

3 KUNNOSSAPITO

Kunnossapidon tärkein tehtävä on pitää käyttöomaisuus kunnossa ja tuottavana koko sen elinkaaren ajan ja mahdollisuuksien mukaan parantaa tuottavuutta ja elinkaarta. Kuvassa 7 on kunnossapidon prosessikaavio.



Kuva 7. Kunnossapidon prosessikaavio. [4]

Kunnossapito määritellään erilaisissa standardeissa seuraavanlaisesti:

SFS-EN 13306: ”Kunnossapito koostuu kaikista kohteen eliniän aikaisista teknisistä, hallinnollista ja liikkeenjohdollisista toimenpiteistä, joiden tarkoituksena on ylläpitää tai palauttaa kohteen toimintakyky sellaiseksi, että kohde pystyy suorittamaan vaaditun toiminnon” [5].

PSK 6201: ”Kunnossapito on kaikkien niiden teknisten, hallinnollisten ja johtamiseen liittyvien toimenpiteiden kokonaisuus, joiden tarkoituksena on säilyttää kohde tilassa tai palauttaa se tilaan, jossa se pystyy suorittamaan vaaditun toiminnon sen koko elinjakson aikana” [6].

3.1 Ennakoiva kunnossapito

Kaikki se kunnossapito, joka suoritetaan ilman, että laitteessa tiedetään olevan vikaa, voidaan laskea ennakoivaksi kunnossapidoksi. Tärkeintä ennakoivassa kunnossapidossa on ehkäistä syntyviä vikatilanteita ja pitää näin ollen laitteen käyttöaste korkeana. Ennakoiva huolto on haasteellinen prosessi missä käyttökertoihin tai aikaperusteisiin laskureihin perustuen laitteelle määritellään huoltorajoja tai huoltovälejä. Ennakoivan ja korjaavan kunnossapidon välille on löydettävissä yleensä järkevä suhde, mikä on taloudellisesti, toiminnallisesti sekä turvallisuusnäkökohdat huomioon ottaen kannattavaa. Valmistusprosessien muuttumisen vähävaiheisemmaksi eli joustavammaksi sekä automaation lisääntymisen myötä tuotantokoneiden hinnat ovat kasvaneet sellaiselle tasolle, että koneen ollessa pois käytöstä välilliset kustannukset ohittavat kunnossapito- ja korjauskustannukset. Tämä luo tiettyjä paineita ennakoivalle kunnossapidolle, joka onkin tämän takia erittäin vakavasti otettava asia.

3.2 Korjaava kunnossapito

Jos ennakoivasta huollosta huolimatta tapahtuu laiterikko, tätä kutsutaan korjaavaksi kunnossapidoksi. Korjaavassa kunnossapidossa on tärkeintä saattaa laite takaisin tuottavaan tilaan mahdollisimman nopeasti kuitenkin korjauksen laadun kärsimättä, jottei vika uusiutuisi. Pienissä vikatapauksissa huoltoa voidaan siirtää myöhempään ajankohtaan esimerkiksi seisakin ajankohdaksi, mikäli siitä ei aiheudu haittaa tuotannolle. Tärkeää on myös huoltomiehen nopea reagointi vikatapauksissa. Samoin varaosien saanti, ja vian analysointi. Joissakin tapauksissa korjaava kunnossapito ei ole enää taloudellisesti ja toiminnallisesti kannattavaa ja näin ollen laite poistetaan käytöstä. Käytöstäpoiston on oltava suunnitelmallista ja organisoitua toimintaa, jotta investoinnista saadaan mahdollisimman suuri hyöty yritykselle. Monissa tapauksissa laitteella on edelleen vielä kaupallista arvoa ja järkevintä on myydä laite eteenpäin.

3.3 Valmistajan käyttörajat

Joissain tapauksessa valmistaja määrittelee tuotteilleen käyttörajat, esimerkiksi kuinka monta käyttökertaa kyseinen osa kestää. Usein kuitenkin nämä käyttörajat on testattu laboratorio-olosuhteissa ja eivät kuitenkaan testiolosuhteissa kestä näin pitkään. Useasti valmistajan käyttörajoja ei ole saatavilla ollenkaan ja näin ollen ne joudutaan itse määrittelemään omien käyttökokemusten mukaan. Näihin käyttörajoihin paneudutaan

tämän opinnäytetyön työsuudessa. Tässäkään tapauksessa valmistajalta ei ollut saatavissa luotettavia spesifikaatioita.

3.4 Parantava kunnossapito

Parantavan kunnossapidon tehtävä on parantaa laitteiden suorituskykyä, turvallisuutta, kestävyyttä tai käytettävyyttä ja vähentää näin korjaavan kunnossapidon tarvetta. Laitteiden modernisointi voidaan laskea myös parantavan kunnossapidon piiriin. Vanhaa laitteistoa voidaan parantaa tai päivittää vastaamaan syntyneitä tarpeita ilman että joudutaan hankkimaan uutta laitteistoa. Parantava kunnossapito on harkittava tapauskohtaisesti sillä useissa tapauksissa vanhan laitteen päivittäminen ei ole pidemmän päälle kannattavaa.

4 KÄYTTÖÖNOTTOPROJEKTIN SUUNNITTELU

Projektimainen toimintatapa on tunnettu jo kauan, mutta vasta viime vuosina se on levinnyt vallitsevaksi tavaksi viedä läpi kertahankkeita yrityksissä. Projektisuunnittelulla on suuri vaikutus koko projektin onnistumiseen. Hyvin läpiviedyssä projektissa kaikki on ennalta suunniteltua, ihmiset on motivoituneita ja projektin tavoitteet on kunnossa. ”Hyvin suunniteltu projekti on puoliksi tehty”. Projektin suunnittelu ei saisi jäädä yhden tai kahden asialle omistautuneen ihmisen varaan, vaan olisi ihanteellista jos projektin suunnitteluun osallistuisi koko projektiryhmä. Hyvä hanke on aina tarvelähtöinen. Ihannelanteessa kehittämistarve nousee esiin yrityksen perustoiminnassa tehdyistä havainnoista. Tällöin projekti ja siitä saadut tulokset muodostuvat luontevaksi osaksi yrityksen toimintaa. [7]

4.1 Projektiehdotus

Projekti alkaa projektiehdotuksesta joka on projektin ensimmäinen vaihe. Tässä vaiheessa projektia määritellään projektin tavoitteet sekä laajuus. Projektiehdotuksessa käsitellään projektin taustatietoja sekä esitetään nykyhetken tilannetta miksi projekti olisi tarpeellinen. Tässä vaiheessa on hyvä myös esittää projektiin sopivia mittareita. Esimerkiksi tulokseen, laatuun ja työaikaan liittyviä mittareita mitkä vaikuttavat yrityksen suorituskykyyn. Investoinnit ja mahdollisesti projektin tuomat säästöt sekä projektin vaikutukset yritykselle sisällytetään projektiehdotukseen. Projektin sisältö on rajattava hyvin sillä se saattaa paisua helposti yli rajojen ja tällöin tulokset eivät välttämättä ole halutun kaltaisia. Projektissa on aina tiedettävä mitä tehdään, kenelle ja miksi. Valitettavan usein projektit epäonnistuvat siitä syystä että ei ymmärretä kunnolla mitä oikeastaan projektissa olisi pitänyt tehdä. [7]

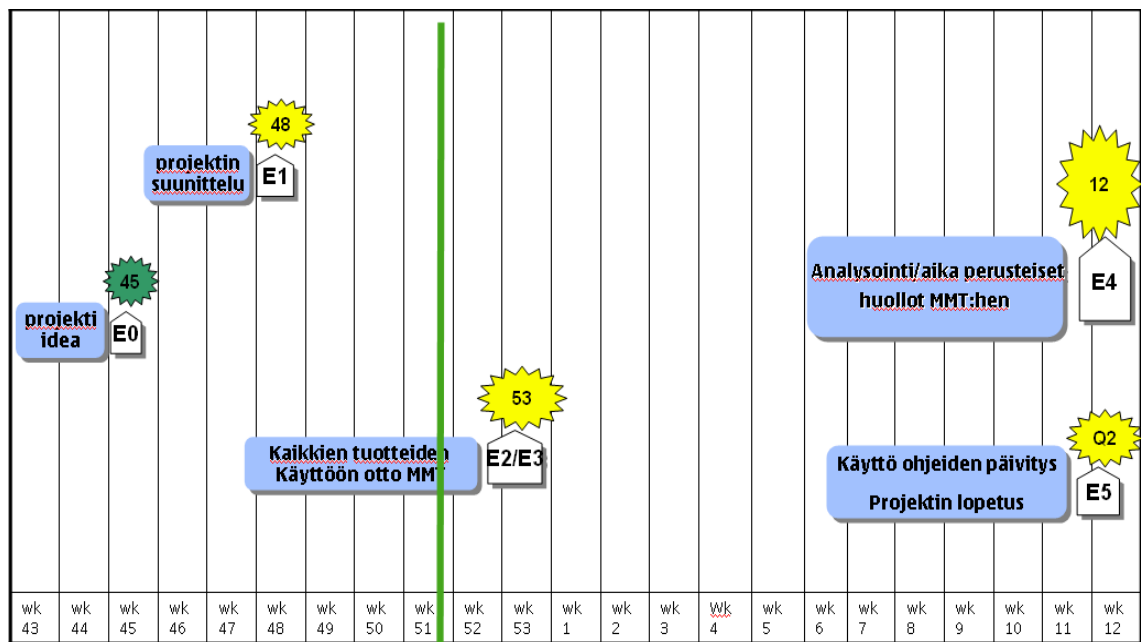
4.2 Projektisuunitelma

Toisessa vaiheessa, kun projekti on hyväksytty tehtäväksi luodaan projektisuunitelma missä määritellään projektin vetäjä ja projektin omistajat. Tiivis paneutuminen tuo tuloksen nopeammin, halvemmalla ja paremmin. Projektisuunitelmassa kuvataan projektia yleisesti, kuvataan projektin tavoitteet ja laajuus. Suunitelmaan sisällytetään projektin mittarit, projektin kustannusarvio, mahdollisesti takaisinmaksulaskelmat, projektiorganisaatio, tehtävät sekä vastuut, projektin sidosryhmät, kommunikointi ja raportointi, muutoshallinta, projektin ositus, projektin riskikartoitus, ympäristö, työturvallisuus, työskentelyolosuhteet, ympäristöasiat, etappikatselmoinnit sekä

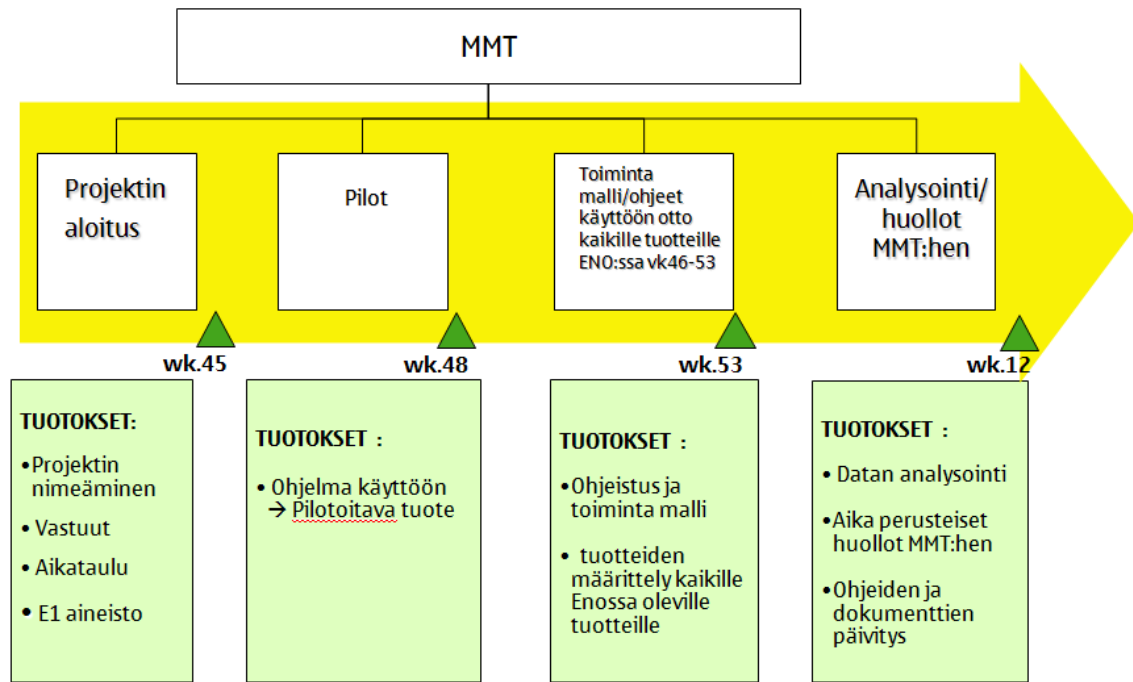
aikataulut. Projektisuunnitelma on tehtävä huolella ja ajan kanssa, sillä jo suunnitteluvaiheessa tehdyt virheet ja laiminlyöntien haitat kostaavat moninkertaisina projektin edetessä. Projektissa ei voi kuitenkaan luottaa pelkästään itseensä vaikka omaisi kuinka paljon tietoa tai kokemusta vaan tarvitaan muiden näkökulmia onnistuakseen. Pelkkä yhden henkilön kokemus antaa yksipuolisen kuvan. [8]

4.3 Etappikatselmoinnit

Etappikatselmoineissa seurataan projektin etenemistä, käydään läpi saatuja tuloksia sekä täsmennetään ja tarkennetaan projektin sisältöä. Kuitenkin lähes aina projekteilla on tapana lähteä rönsyilemään ja selkeäkin tavoite saattaa muuttua epäselväksi projektin edetessä. Sisällön, kustannuksien sekä aikataulumuutoksien vaatimat asiat on syytä siirtää jatkoprojekteihin, jollei niistä luovuta kokonaan. Kaikkia muutoksia ei voida kuitenkaan välttää sillä esimerkiksi projektin tarpeet saattavat muuttua matkan varrella. Kuvissa 8 ja 9 kuvataan tämän kyseisen projektin ajanhallintaa.



Kuva 8. Projektin etapit.



Kuva 9. Projektin aikataulu.

Projektit voidaan viedä läpi erilaisin menetelmin. Useasti yrityksillä on määritelty tapa projektin suorittamiseen. Projektin aikatauluksessa käytetään yleisesti myös janakaavioita ja tehtävä verkostoja. Vaikka projekti jaettaisiin etappeihin, silti etapeiden välissä kannatta pitää välipalavereita.

4.4 Ajankäyttö

Usein projektiin osallistuvat ihmiset saattavat työskennellä myös muissa projekteissa tai ovat projektissa mukana oman työnsä lisäksi. Projektisuunnitelmassa on hyvä myös määritellä työntekijöiden ajankäyttö, jonka he käyttävät projektiin. Taulukossa 1 on kuvattu ajankäyttösuunnitelmää yksinkertaisella taulukolla.

Taulukko 1. Ajan käyttö.

Henkilö	Työtehtävä	Ajankäyttö työajasta
Herra x	Huoltoteknikko	20 %
Neiti y	Sihteeri	30 %
Herra z	Insinööri	40 %

Ihannetilanteessa projektiryhmä hoitaisi vain yhtä projektia ja voisi paneutua asiaan kokoaikaisesti. Päällekkäiset projektit huonontavat tulosta sillä keskittyminen häiriintyy, mistä seuraa todennäköisesti huonompi lopputulos, hitaammin ja suuremmin kustannuksin. Silti tänä päivänä kustannustehokkaan liiketoiminnan ja rajallisten resurssien paineessa on yleinen tapa hoitaa monta projektia yhtä aikaa. Myös tämä kyseinen projekti hoidettiin omien normaalien töiden ohella ja tämä muodosti oman haasteensa projektin läpiviemisessä. [7]

4.5 Riskikartoitus

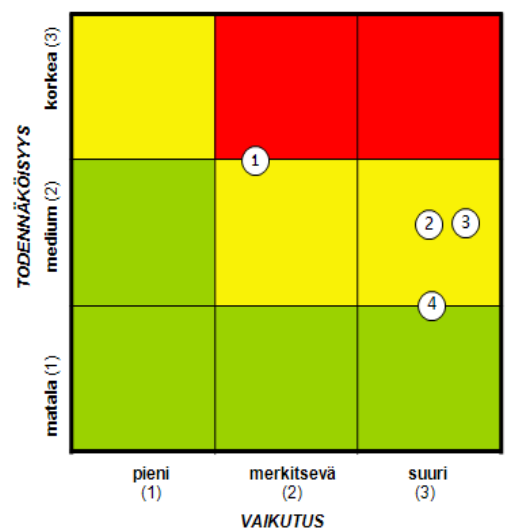
Projektisuunnitelmaan sisältyy yleisesti myös riskikartoitus, jossa käsitellään projektin riskejä ja mahdollisesti niitä ehkäiseviä toimenpiteitä. Riskikartoitusta päivitetään projektin edetessä ja näin ollen myös uusien riskien ilmaantuessa. Arvioidaan parhaaksi katsottujen vaihtoehtojen riskit. Arvioidaan niiden mahdollisuus ja vakavuus sekä toimenpiteet riskien pienentämiseksi tai ehkäisemiseksi. Kuvassa 10 on tämän projektin ensimmäinen riskianalyysi.

Riskit

- 1) Datan kerääminen/analysointi ohjelmassa.
Kunnollinen työkalu ohjelmaan?
- 2) Raportointi – saadaanko vuorohuolto käyttämään ohjelmistoa?
- 3) Ohjelmiston bugit ?
- 4) Resurssit millä aikataululla tuotteet saadaan kantaan ja ohjelmistoon

Ehkäisevät toimenpiteet

- 1) Ajoissa yhteyttä ohjelman suunnittelijaan
- 2) Uutta mallia suunniteltaessa otettava huomioon - ohjeistus/koulutus. Käyttöopastusta kun ohjelma käytössä. Lisää toimintoja mistä vuorohuollolle hyötyä päivittäisessä käytössä.
- 3) Etukäteen ohjelmaan tutustuminen
- 4) ?



selite

riskin numero ja asema kartalla
arvioitu riskin siirtyminen
riskin viimeisin liike



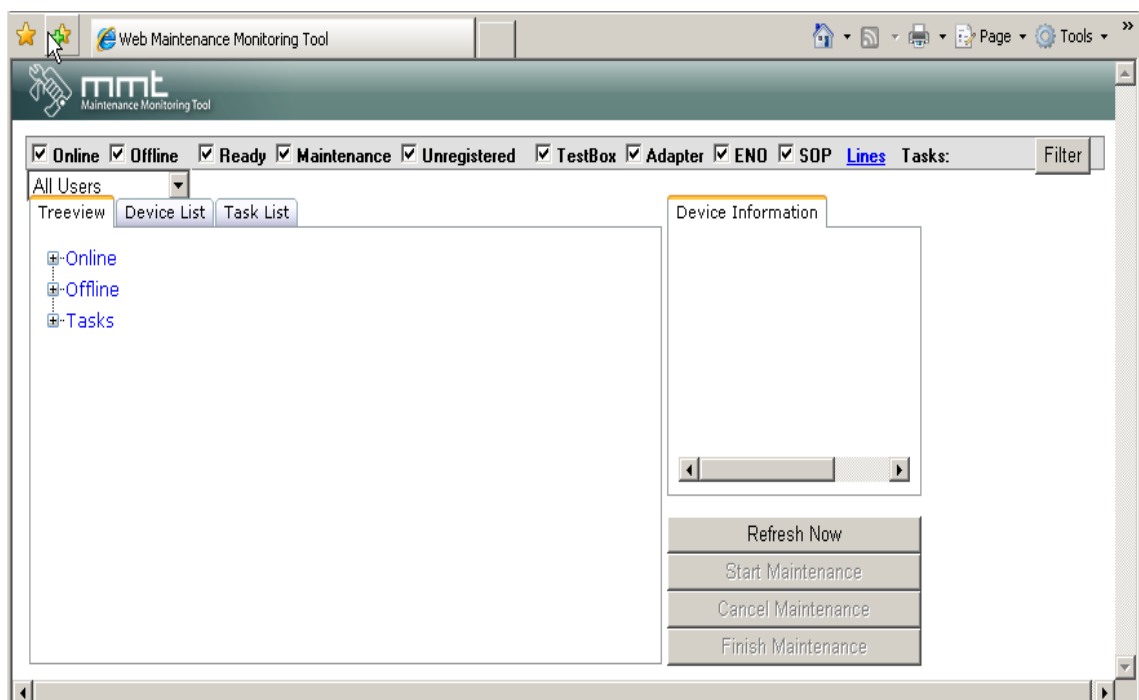
Kuva 10. Projektin ensimmäisen etapin riskianalyysi.

4.6 Investoinnit

Useasti projektit vaativat kustannuslaskelman. Kustannuslaskelmat on syytä tehdä huolella. Kustannuslaskelmissa huomioon otettavia asioita ovat esimerkiksi: palkkakulut, asiantuntijapalkkiot, matkakulut, laitehankinnat, vuokratkustannukset, toimintatulot, EU-rahoitus, valtion rahoitusosuus, kuntarahoitus, muu rahoitus ja muut kulut. MMT projekti suoritettiin ilman ulkoisia hankintoja mikä helpotti projektin läpiviemistä merkittävästi.

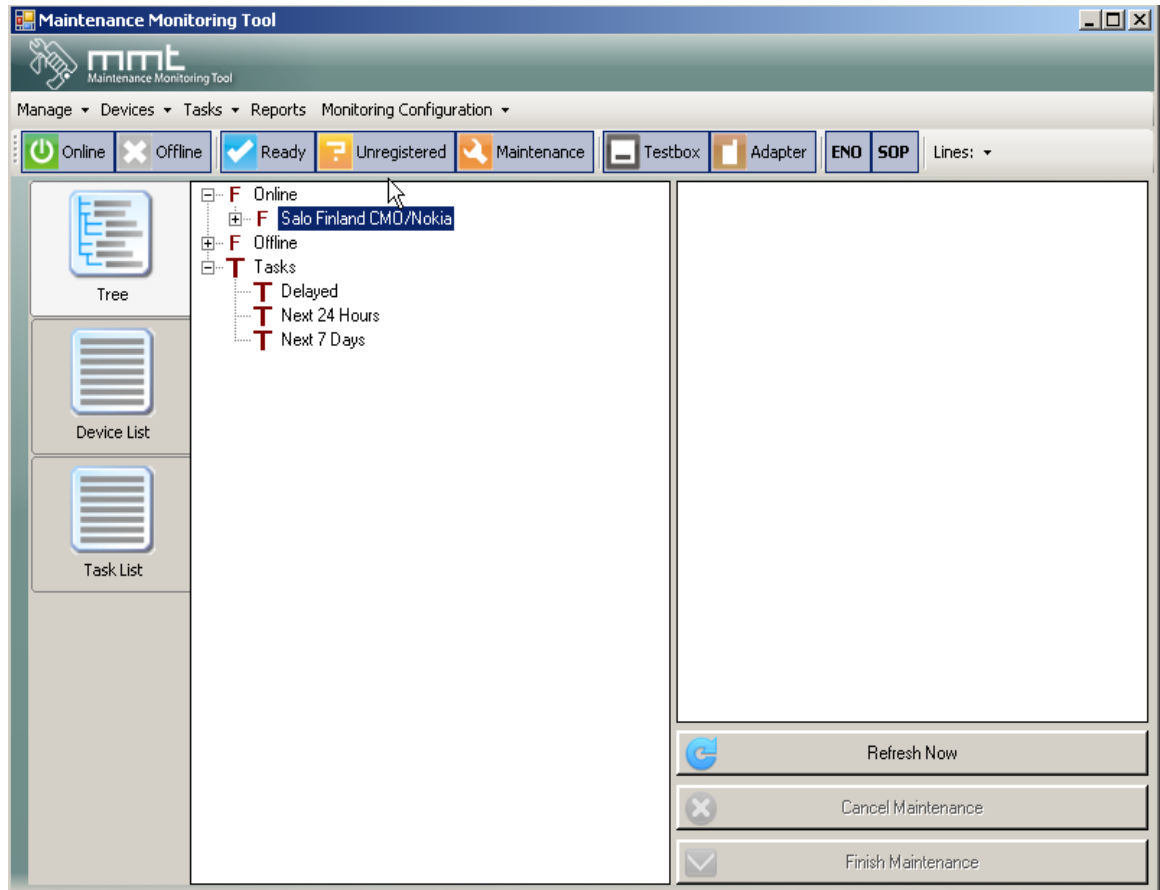
5 MAINTENANCE MONITORING TOOL - KÄYTTÖÖNOTTOVAIHE

Maintenance Monitoring Tool on työkalu huoltojen, raportointien ja tiedon analysoimiseen. Ohjelmisto on kehitetty vain kohdeyritystä varten eikä ole julkisessa levityksessä. Työkalu tehtiin mahdollisimman riippumattomaksi ympäristöstä. Ohjelmisto käyttää testilaitteiden paikallistamiseen avuksi tietokantoja. Se mahdollistaa tuotannon seurannan ja hallinnan sekä tuotteiden jäljitettävyyden tuotannon aikana. Työkalussa on kaksi erillistä käyttöliittymää raportointisovellus ja varsinainen järjestelmän ylläpityökalu. Kuvassa 11 on raportointisovelluksen käyttöliittymä.



Kuva 11. Raportointisovellus.

Raportointisovellus on lähinnä huoltomiesten käyttöön suunniteltu selainpohjainen työkalu jota pystyy käyttämään jokaiselta tietokoneelta työpaikan tietoverkossa. Kaikki raportointi on tarkoitettu hoitaa raportointisovelluksesta ja ylläpityökalu on suunniteltu vain järjestelmän ylläpitäjien sekä työnjohdon käyttöön. Ylläpityökalu sisältää testilaitteiden määrittelyt, huoltorajat, aikaperusteiset huollot, raportit ja käyttäjätunnukset. Ylläpityökalussa on myös mahdollisuus määrittellä hälytyksiä käyttäjille. Työkalu lähettää automaattisesti sähköpostia määritellylle henkilölle, kun testilaitte vaatii huoltotoimenpiteitä. Kuvassa 12 on ylläpityökalun käyttöliittymä.

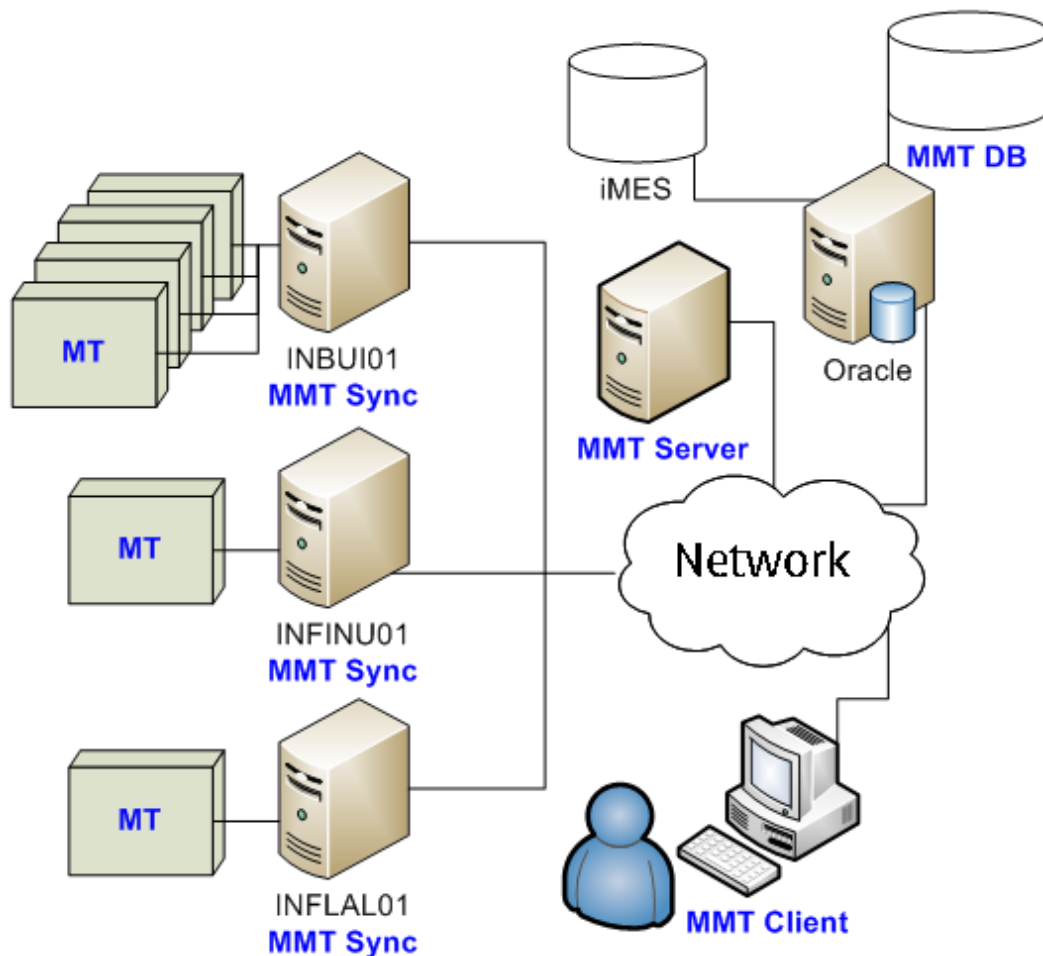


Kuva 12. Ylläpitotyökalu.

5.1 MMT-toimintaperiaate

MMT toimii palvelimelta käyttäen avuksi yrityksen tietokantoja. Yksinkertaistettuna ohjelma lukee sarjanumeron adapterin ja testausmekaniikan muistipiiriltä. Ohjelmassa luodaan kortteja eri tuotteille, jotka sisältävät erilaisia huolto- ja korjauskohteita. Ohjelma laskee käyttökertoja testeistä ja näiden mukaan määrittellään huoltovälit analysoidun datan perusteella.

Kuvassa 13 on MMT:n toimintaperiaate. MMT DB kerää ja lähettää Oraclen relaatiotietokantaan syötetyn tiedon talteen kuten linjan, vaiheen, sijainnin, testimekaniikan, käytetyn adapterin ja osat. Testilaitteiden tiedot syötetään tietokantaan iMES käyttöliittymän kautta. MMT sync, joka kerää tiedon mekaaniselta testilaitteelta (MT=mechanical test device) ja lähettää tiedon MMT-palvelimelle. Ohjelmistoa varten yrityksessä on oma palvelin MMT-työkalulle. MMT Clientissa tapahtuu huoltojen tekeminen, käyttäjien, osien ja hälytysviestien hallinnointi, laitteiden määrittelyt, aikaperusteiset kortit sekä raportit



Kuva 13. MMT toimintaperiaate.

5.2 Projektin alustus

”Projektin on väliaikainen organisaatio, joka puretaan kun tavoite on saavutettu.” [7] Tässäkin tapauksessa muodostettiin väliaikainen organisaatio eli lyhytaikaista ongelman ratkointia varten ei tarvitse varata ylimääräisiä resursseja vaan projekti voidaan hoitaa operatiivisten työtehtävien ohella. Tämän projektin tarpeellisuus on muodostunut työnantajan tarpeesta kehittää ennakko- ja huoltotoimintaa tehokkaammaksi vastaamaan nykypäivän tehokkuusvaatimuksia. Projektissa on pystyttävä työskentelemään tehokkaasti ja sovitussa aikataulussa. Lisäksi myös sillä miten hyvin saa tukea muilta silloin, kun siihen on tarvetta, on suuri merkitys projektin onnistumiseen.

5.3 Projektin tavoitteet

Projektin tavoitteeksi määriteltiin ohjelmiston käyttöönotto LENO-alueella. Projekti sisältyi Flali- ja Finui-testausta. Projektin tavoitteisiin kuului saatavan datan hyödyntäminen, toimintamalli, käyttöohjeiden luominen, huoltovälien määrittäminen ja aikaperusteisten huoltojen siirtäminen eri ohjelmasta MMT työkaluun. Projekti alkoi projektisuunnitelman laatimisella. Yritys käyttää projektien läpiviemiseksi kuusiportaista etappeihin perustuvaa projektimallia. Projektimallin ensimmäinen etappi 0 oli projektin ehdotus mikä jäi pois MMT-projektista, koska projekti oli päätetty jo toteuttaa. Projektisuunnitelman ensimmäinen vaihe tässä tapauksessa oli etappi 1:n luominen. Käytännössä määriteltiin projektin aikataulutus sekä projektin sisältö. Ensimmäinen etappi hyväksyttiin projektin omistajilla ja näin projekti pääsi varsinaisesti alkamaan. Tässä vaiheessa ei listattu vielä liikaa tavoiteltavia asioita vaan itse käyttöönotto oli projektin varsinainen tavoite. Projektin etappi 2 ja 3 sisälsivät kaikkien tuotteiden käyttöönoton. Etappi 4 sisälsi tulosten analysoinnin ja aikaperusteisten huoltojen siirtämisen työkaluun. Etappi 5 oli käyttöohjeiden päivitys ja projektin lopetus. Näkökantoina käytettiin yrityksen muiden työntekijöiden sekä allekirjoittaneen näkökulmia. Projektia suunniteltaessa ei tule unohtaa kirjallisuuslähteitä joita on saatavilla suuret määrät. Näistä saa paljon apua projektin läpiviemiseen ja lisäksi kirjallisuudesta löytyy suuret määrät eri ongelmakohtia mitä projektin edetessä saattaa kohdata.

5.4 Yrityksen vaatimukset ohjelmistolta

Ennakkohuoltojärjestelmää valittaessa tärkeintä on sopivuus ja liitettävyys tuotantoprosesseihin. Lisäksi järjestelmän tulisi olla helppokäyttöinen, jotta ei tulisi paljoa uutta opiskeltavaa ja järjestelmän käyttö olisi sujuvaa. Tässä tapauksessa yritys päätyi kehittämään oman järjestelmän jota voidaan räätälöidä tarpeen mukaan ilman lisenssimaksuja. Ongelmaksi tässä tapauksessa muodostui ohjelmistokehitysyksikön resurssit sekä ohjelmiston kehityksen siirtyminen eri henkilöiden tehtäväksi loppuvaiheessa käyttöönottoprojektia. Tämä puolestaan viivästytti ohjelmaan tarvittuja parannuksia.

5.5 Ohjelmiston testaus

Projektin toteutus lähti käyntiin ohjelmiston testauksella. Tavoitteena oli ohjelman ominaisuuksien ja toiminnallisuuksien testaaminen. Ohjelma tuli tuotantoon aluksi määriteltyjen vaatimuksien pohjalta joita lähdettiin räätälöimään tuotantoon sopivaksi. Näin ollen kyse oli ohjelmiston käyttöönottovaiheen toiminnallisuustestauksesta.

Ohjelman ominaisuuksien ja toiminnallisuuden lisäksi tässä vaiheessa testattiin myös ohjelman kahta käyttöliittymää ja niiden toimintaa. Käyttöliittymistä puuttui paljon tärkeitä yksityiskohtia jotka tulivat ilmi vasta käytössä. Projektin alkuvaiheessa sovittiin, että jokainen huoltotoimenpide merkitään työkaluun. Kaikki huoltotoimenpiteet eivät kuulu ennakkohuoltotoiminnan piiriin, mutta tätä kautta mahdollisesti usein vaihdettavia osia on mahdollisuus sinne lisätä.

5.6 Tuotteiden konfigurointi

Projekti alkoi käytännössä eri tuotteiden huoltokorttien tekemisellä ohjelmaan. Jokaiselle tuotteelle luotiin eri testivaiheelle omat kortit joihin kirjattiin adapterin ennakkohuollon piiriin kuluvat osat. Ennakkohuollon piiriin kuuluvia osia ovat kaikki liikkuvat sekä kuluvat osat. Esimerkiksi testineulat, liittimet (USB, HDMI, DC, audio), liikkuvat testausjohtimet ja mittapäät. Osista löytyi varsin kattavat manuaalit ja osat käytiin läpi tuotteesta vastaavan testausinsinöörin kanssa. Samalle tuotteelle jouduttiin tekemään jopa kolme eri korttia koska adapterin muistit poikkesivat toisistaan ja ohjelma ei näin tunnistanut adapteria samaksi tuotteeksi jolle kortti löytyi. Alkuvaiheessa tuotekortit eivät olleet vielä täydellisen kattavia vaan tärkeämpää oli, että ohjelmisto saataisiin mahdollisimman nopeasti käyttöön, ja näin kehittämään

toimivammaksi. Jatkossa uutta tuotetta konfiguroitaessa osat määritellään tuotteesta vastaavan insinöörin kanssa. Monet tuotteet käyttävät samoja osia adaptereissaan ja näin ollen aikaisempia huoltorajoja voidaan käyttää heti uudella tuotteella.

Jo alkuvaiheessa ohjelmasta löytyi vakavia puutteita joista korjauspyynnöt lähtivät välittömästi eteenpäin. Tuotteiden konfigurointi aloitettiin ensimmäiseksi tekemällä yhdelle tuotteelle sekä Flali että Finui kortit ja näin ohjelma saatiin koekäyttöön. Myös mekaniikoille tehtiin omat kortit jo alkuvaiheessa. Tässä vaiheessa paljastui, että ohjelma ei toiminut käytössä olevalla internetselaimella kunnolla ja näin ollen jokaisen testilaitteen PC:lle jouduttiin lisäämään toinen internetselain.

5.7 Tilastollisen aineiston hyödyntäminen ja ennakkohuoltorajat

Ohjelma keräsi tietoa alusta asti huolloista, mutta raportit olivat toistaiseksi käytännössä käyttökelvottomia. Esimerkiksi raportista saattoi löytyä kellonaika, mutta ei ollenkaan päivämäärää siitä, milloin huolto oli tehty. Raportteja pyrittiin kehittämään paremmaksi antamalla palautetta ohjelman tekijöille. Raportointityökalussa on vapaatekstikenttä jota esimerkiksi ei aluksi saanut tulostetuksi. Raportteihin saatiin muutos suhteellisen nopealla aikataululla, ja näin päästiin seuraamaan laajemmalti raportointeja ja ohjelman käyttöä. Kertyvää dataa oli tarkoitus analysoida projektin loppuvaiheessa kun tarpeeksi tietoa olisi kertynyt. Raporttien kehitys on jatkunut koko projektin ajan ja vaativat edelleen kehittämistä projektin päättyessä. Projektin loppuvaiheessa ennakkohuoltorajoja päästiin jo määrittelemään.

Ennakkohuoltorajojen määrittelyssä kaikkien osien vaihtorajat asetettiin äärettömän suuriksi. Käytännössä tämä tarkoitti, että mitään osia ei vaihdettu laskureiden mukaan, vaan ohjelmaan raportoitiin todellisen vikaantumisen johdosta tehdyt toimenpiteet. Projektin loppupuolella rajoja alettiin määritellä kertyneen datan perusteella. Rajat määriteltiin aluksi hieman tavoitteita korkeammiksi ja apuna käytettiin myös edellisen huoltolaskurin rajamäärityksiä. Kertyneestä datasta havaittiin osien keston vaihtelevan melkoisesti eikä näin lyhyellä seurannalla kovin tarkkoja rajoja pystytä määrittelemään. Rajamääritykset täytyy harkita muutoinkin todella tarkkaan, koska liian pienet vaihtorajat nostavat varaosakustannuksia merkittävästi. Ennakkohuolto on osa koko yrityksen systemaattista toimintaa. Sitä ei pidä hoitaa erillisenä toimintona muusta kunnossapidosta, vaan sen pitäisi sisältyä päivittäisiin toimiin. Ennakkohuolto on erittäin tärkeä toiminto yrityksessä mutta jää monesti liian vähälle huomiolle.

Taulukossa 2 havaitaan vertailemalla minimi- ja maksimilukemia keskimääriseen vaihtoväliin osien kestävyiden vaihtelun olevan erittäin suuri.

Taulukko 2. Rajojen määrittäykset.

Changes	Min Life Cycle	Max Life C	Average Life Cycle
54	261	19163	4959.333333
6	160	10468	5612
29	10	23327	4737.310345
28	649	22308	8410.714286
2	10468	22399	16433.5
18	2732	19605	10722.33333
Changes	Min Life Cycle	Max Life C	Average Life Cycle
29	0	58360	4075.931034
26	0	14879	2085
14	1123	44469	8479.928571
5	1342	44469	15960.4
5	3261	44469	17168.4
9	12916	44469	24490
Changes	Min Life Cycle	Max Life C	Average Life Cycle
7	933	9816	3538.857143
1	1728	1728	1728
2	1571	19805	10688
Changes	Min Life Cycle	Max Life C	Average Life Cycle
24	15999	193761	120381
15	871	67648	13422.8
2	4905	26588	15746.5
16	0	193761	44776.375
12	2096	186895	76259.83333
10	10585	193761	88958.6
24	76961	193761	156737.5
Changes	Min Life Cycle	Max Life C	Average Life Cycle
41	0	36862	3086.829268
34	0	36862	3004.382353
11	168	42944	12234.54545
10	20	37624	8727.4
25	1472	39459	15033.16
13	0	44906	9307.384615

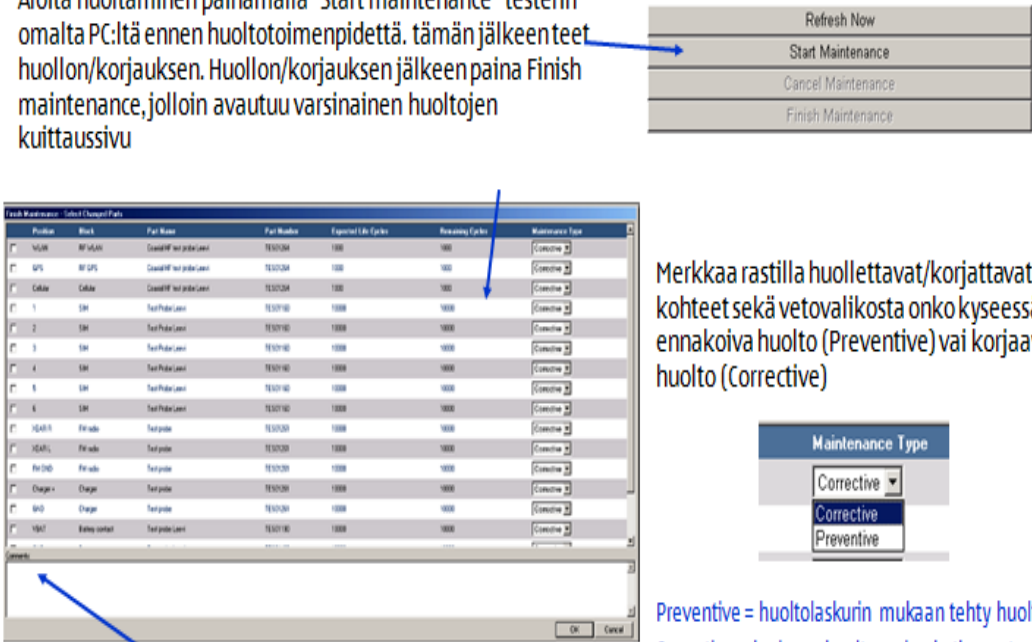
Rajoja tullaan korjamaan jatkossa varmasti useaan otteeseen, kunhan päästään seuraamaan osien kestävyttä pidemmällä aikavälillä. Tämä tulee kuitenkin olemaan pitkä prosessi, joka vaatii aktiivista seurantaa.

5.8 Käyttöopastus

Ohjelmistosta löytyi alussa käyttöohjeet molempiin käyttöliittymiin. Käyttöohjeita päivitettiin projektin aikana useaan otteeseen. Myös yrityksen tietokantaan tallennettiin työkalun käyttöohjeet, jotta ne olisivat sieltä saatavissa. Ohjelmistosta järjestettiin koulutukset jo ennen käyttöönottoprojektin alkamista. Projektin aikana annettiin myös henkilökohtaista opastusta ohjelmaan, mikäli tarvetta ilmeni. Ohjelman käytölle laadittiin erillinen työkulkukaavio siitä miten raportointi oli tarkoitus hoitaa. Kuvassa 14 on vuorohuollon raportoinnin pikaohje.

Huoltojen kirjaaminen

Aloita huoltaminen painamalla "Start maintenance" testerin omalta PC:ltä ennen huoltotoimenpidettä. tämän jälkeen teet huollon/korjauksen. Huollon/korjauksen jälkeen paina Finish maintenance, jolloin avautuu varsinainen huoltojen kuitaussivu



Merkkaa rastilla huollettavat/korjattavat kohteet sekä vetovalikosta onko kyseessä ennakoiva huolto (Preventive) vai korjaava huolto (Corrective)

Preventive = huoltolaskurin mukaan tehty huolto
Corrective = korjaava huolto, esim. katkennut testiipiikki

Huollon lopuksi merkitse Comments kenttään mistä huolto-/korjaustoimenpide aiheutunut

Huom! Käyttämällä ennen ja jälkeen huollon "Start maintenance" ja "Finish maintenance" näppäimiä, saadaan järjestelmään aikaleima, jolla päästään seuraamaan historiatiedoista huoltojen kestoja

Kuva 14. Vuorohuollon raportoinnin pikaohje.

5.9 Aikaperusteisten huoltojen siirtäminen työkaluun

Aikaperusteisten huoltojen siirtäminen työkaluun katsottiin myös tarpeelliseksi, jotta huoltomiesten ei tarvitsisi käyttää työssään monta työkalua huoltojen kuittaamiseksi. Viikko- ja kuukausihuollot siirrettiin työkaluun muuttamalla ne käyttökertaperusteisiksi. Huolloille määriteltiin kappalemääräiset rajat, joiden mukaan huollot suoritetaan. Tämä muutos säästää aikaa ja poistaa turhaan suoritettavia huoltotoimenpiteitä, koska esimerkiksi laskurin mukaan lähellä oleva huolto voidaan suorittaa testilaitteen vikaantumisen yhteydessä. Ainoastaan yksi huoltotoimenpide vaati aikaperusteisen laskurin käyttöä. Tämä huoltotoimenpide oli testereiden tietokoneiden suodattimien imurointi, joka siirrettiin koontaoperaattoreiden tehtäväksi viikkosiivouksen yhteyteen. Imuroinnin siirtämisen yhteydessä imuroinnille jouduttiin suorittamaan ESD-mittaukset, koska jatkossa imurointi suoritetaan irroittamatta suodatinta. ESD-mittauksissa ei havaittu syytä, miksi ei imurointia voitaisi näin suorittaa. Tässä vaiheessa ideoitiin myös muiden kuin testauslaitteiston ennakkohuoltotoiminnan siirtämistä työkaluun. Tämä koskisi jatkossa esimerkiksi automaattiruuvaimia, koontajigejä ja ruuvausautomaatteja. Tämä vaatii kuitenkin laskureiden ohjelmistopäivityksen, ja tästä on esitetty myös kehityspyyntö ohjelman tekijöille.

6 KEHITYSIDEAT JA PALAUTE

Uutta työkalua voidaan verrata aikaisempaan huoltolaskurijärjestelmään joka toimi samalla periaatteella kuin MMT-työkalu. Ennakkohuoltolaskuri on tuotannossa käytössä oleva laitekohtaisia testauskertoja ja paikkatietoja seuraava ohjelma. Vanhan järjestelmän kankeus on aiheuttanut sen, että laskurit olivat jatkuvasti väärin evätkä siis ajan tasalla, koska ohjelma vaati käyttäjältään jokaisen paikkatiedon muutoksen tehtäväksi manuaalisesti. Uusi ohjelma toimii automaattisesti, kun siihen on kerran määritelty laite niin ohjelma osaa automaattisesti hakea tiedot järjestelmästä mikäli testilaitteen paikka vaihtuu. Kuten aiemminkin raportointi tapahtuu Intranetissä sijaitsevan käyttöliittymän kautta.

6.1 Palaute

Ohjelmisto oli määritelty kehitysvaiheessa hyvin pitkälle mitä ohjelmalta vaaditaan. Uutta ohjelmaa suunniteltaessa on erittäin vaikea ottaa huomioon kaikkia näkökantoja ja olisi ensiarvoisen tärkeää ottaa suunnitteluun mukaan ohjelman loppukäyttäjät. Paras palaute tulee yleensä käyttäjiltä ja osan tarpeista huomaa vasta käytössä. Suurin osa ohjelman jatkokehityksestä tapahtui omien havaintojen sekä huoltomiesten käytössä esiintulleista ongelmista sekä näiden kautta syntyneistä parannusideoista. Parannus- ja kehitysideoita kerättiin lähinnä sähköpostin sekä suullisesti saadun palautteen avulla. Ohjelmistolla oli kaksi pääkäyttäjää jotka keräsivät palautetta kentältä. Kaikki palaute raportoitiin eteenpäin välittömästi ohjelmankehittäjille.

6.2 Varaosamerot

Projektin alkuvaiheessa huomattiin varaosameroiden puuttuvan työkaluista. Yrityksen jokaisella varaosalla on oma varaosakoodi, jonka mukaan ne ovat lajiteltu hyllyyn. Varaosanumeron lisääminen helpotti merkittävästi varaosien hakua ja löytämistä vikatilanteessa. Tämä päivitys saatiin työkaluun suhteellisen nopeasti. Jatkossa varaosakoodin yhteydessä voisi mahdollisesti olla jopa hyllypaikan numero, josta varaosan löytää.

6.3 Adapterien muistit

Ohjelman käyttöönottovaiheessa testilaitteiden muisteissa huomattiin olevan erillaisia konfigurointikäytäntöjä. Muutamia muisteja konfiguroitiin uudelleen, että saatiin kaikki

testilaitteet näkyviin ohjelmassa. Jatkossa kaikki adaptereiden muistit tullaan tarkastamaan uusien tuotteiden saapuessa tuotantoon. Mikäli adapterin muistissa tuotteen nimessä on pienikin eroavaisuus joudutaan MMT-työkaluun konfiguroimaan jokaista nimeä kohti oma laitekortti. Tarvittava laitteisto on saatavilla jo yrityksessä. Adapterin muistin tarkistaminen on erittäin helppo toimenpide ja säästää melkoisesti aikaa uutta tuotetta konfiguroitaessa MMT-työkaluun.

6.4 Raportointiliittymän hitaus ja online tila

Ohjelmiston käyttöönoton alkuvaiheessa huomattiin web-clientin olevan todella hidas Internet Explorer -selaimella. Tämän johdosta kaikille testilaitteille asennettiin Mozilla Firefox -selain, koska ohjelma toimi huomattavasti nopeammin tällä selaimella.

Testilaitteelle on määritelty ohjelmaan aika kuinka kauan se pysyy online tilassa, kun testilaitte ei testaa puhelimia. Testilaitetta korjattaessa laite siirtyi liian nopeasti offline tilaan mikä puolestaan hidastutti laitteen uudelleen paikallistamista huoltoa lopetettaessa. Tämä aika muutettiin pidemmäksi. Huoltokohteen paikallistamiseksi tehtiin lisäksi ohjelmaan muutos jotta ENO- ja ATOalue saadaan jaoteltua napin painalluksella.

6.5 Vuorotiedon näkyminen työkalussa

Työnjohdolta saaduissa palautteissa ohjelmaan pyydettiin myös käyttäjän vuorotiedon tallentamista työkaluun. Raportteja seurattaessa saatu tieto päästään lajittelemaan oman vuoron mukaan mikä taas helpottaa vuoron työnjohtajan käyttöseuraamista. Tämäkin ominaisuus on kehityslistoilla ja tulossa jatkossa työkaluun.

6.6 Laiteluettelonäkymä

Laiteluettelonäkymässä raportointisovelluksessa on helppo paikallistaa punaisella oleva eli huoltoa vaativa kohde. Haittapuolena laiteluettelonäkymässä on, että laitteen sijaintia ei näe suoraan, vaan laitteen joutuu paikallistamaan sarjanumeron perusteella. Sarjanumeron perusteella adapterin paikallistaminen on hankalaa ja aikaa vievää. Tämä ongelma tullaan korjaamaan jatkossa päivitysten yhteydessä. Huoltomiehillä asennettiin väliaikaisesti yhdelle koneelle ylläpitotyökalu huoltojen paikallistamiseksi, koska ylläpitotyökalu näyttää myös laitteen sijainnin automaattisesti. Kuvassa 15 on laiteluettelonäkymä.

Online Offline Ready Maintenance Unregistered TestBox Adapter Lines Tasks: All Users

Treeview Device List Task List

LEEVI_FLALI	10748637	128477	980725	View
LEEVI_FLALI	10748634	407107	937488	View
LEEVI_FLALI	10748635	103797	942604	View
DCT4	10724261	970282	994091	View
DCT4	10730684	336487	999284	View
DCT4	10724262	1008632	997818	View
DCT4	10725950	614750	999513	View
DCT4	10707498	1278357	992968	View
DCT4	10725952	828972	992969	View
DCT4	10725955	713750	995283	View
DCT4	10725954	810366	994246	View
STELLA_FLALI	10752849	160746	986820	View
STELLA_FLALI	10752848	20423	988399	View
DCT4	10703019	1297398	985913	View
DCT4	10703018	1412051	983799	View
STELLA_FINUI	10752841	102234	978079	View
STELLA_FINUI	10752840	167239	978175	View
STELLA_FINUI	10752843	91146	984164	View
DCT4	10724486	542366	995000	View
STELLA_FINUI	10752845	110376	972320	View
STELLA_FLALI	10752847	160745	948420	View

Device Information
STELLA_FLALI(10752849)
 Device Type: Adapter
 SN: 10752849
 Remaining Cycles: 986828
 Usage: 1
 Current Cycle: 160746
 Stage: Good
 State: Ready
Parts
 Micro SD
 [1] TES01261 (6200)
 [2] TES01261 (6200)
 [1] TES01261 (6200)
 [2] TES01261 (6200)
 [3] TES01261 (6200)
 [4] TES01261 (6200)
 [5] TES01261 (6200)
 [6] TES01261 (6200)
 Stella Clean
 [Clean] C1 ()
 FM Radio
 [XEAR_L] TES01261 (6200)
 [FM_GND] TES01261 (6200)
 [XEAR_R] TES01261 (6200)
 Cellular
 [1] TES01264 (6244)
 Trace & Flash
 [1] TES01261 (6200)
 Communication

Refresh Now
 Start Maintenance
 Cancel Maintenance
 Finish Maintenance

Kuva 15. Laiteluettelonäkymä.

7 ARVIONTI JA POHDINTA

Ohjelmisto tulee vaatimaan varmasti kehitystä vielä pitkään. Tässä vaiheessa kehitysideoita on syntynyt jo runsaasti ja niiden toteuttamiseen kuluu varmasti aikaa. Uusia ideoita syntyy jatkuvasti ja käytön laajetessa koko yritykseen niitä syntyy varmasti jatkuvasti lisää. Kaikki tähän mennessä syntyneet kehitysideat on raportoitu jo eteenpäin projektin edetessä. Silti ohjelmisto on jo käyttöönoton alkuvaiheessa paljon monipuolisempi ja toimivampi kuin yksikään aikaisemmista ennakkohuolto-ohjelmistoista, joita yrityksellä on ollut käytössä.

7.1 Tulokset

Alussa määritelty tavoite oli ottaa työkalu käyttöön LENO alueella. Ohjelman käyttöönotosta laadittiin projektisuunitelma jonka mukaan edettiin. Projekti eteni koko kestdnsa ajan täysin aikataulussa ja alkuun määrättyt tavoitteet toteutuivat. Jo projektin alkuvaiheessa oli tiedossa, että huoltorajamäärityksiä ei pystytä tällä aikataululla luotettavasti tekemään vaan se vaatii huomattavasti pidemmän seuranta-ajan. Projektin aikataulutukseen sisältyi suuri riski, koska käyttöönottoprojekti oli riippuvainen ohjelmiston kehittäjistä. Silti pakolliset korjaukset onnistuttiin hoitamaan työkaluun projektin aikataulutuksen mukaisesti ja työkalu saatiin käyttöön ajallaan.

7.2 Työn ja menetelmien arviointi

Projektin suorittaminen tapahtui oman työn ohella ja näin ollen etukäteen oli todella hankala arvioida tulevaa työkuormaa, joka asetti oman haasteensa projektin suorittamiselle. Itse projekti vietiin läpi yrityksen toimintatavan mukaan etappeina, joita ei lähdetty kyseenalaistamaan. Jälkeenpäin ajateltuna olisi ehkä kuitenkin ollut järkevää pohtia myös muita projektimalleja vaikka tämä menetelmä tuntuikin tehokkaalta. Muista projektimalleista olisi mahdollisesti saanut uusia näkökulmia projektityöskentelyyn. Ohjelman käyttöönottoprojekti kestoksi määriteltiin 6 – 8 kuukautta. Tämä aika oli ehdottomasti liian lyhyt esimerkiksi huoltorajojen määrittelyn suhteen. Huoltorajojen määrittelemisestä riittäisi varmasti tekemistä helposti ainakin yhden opinnäytetyön verran.

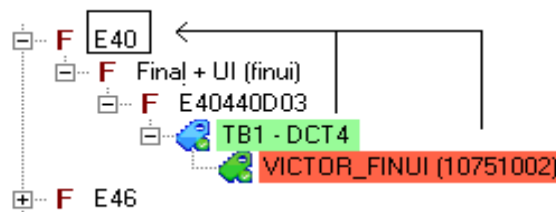
7.3 Työn tulosten luotettavuus ja jatkojalostus

Työn tulokset ovat sinänsä luotettavia mitä käyttöönottoon tulee ja kaikki kehitysideoita on perusteltuja ja ohjelmistoa parantavia asioita. Huoltorajojen määrittelyyn käytetyn datan oikeellisuudesta vastaa ainostaan huoltomiehet. Ohjelmasta puuttuu paljon raportointeja mikä vääristää osista saatavia vaihtovälejä. Sen vuoksi onkin rajamäärityksiin suhtauduttava erittäin kriittisesti. Myös tämä asia on saatettu huoltomiesten tietoon ja sieltä saatava palaute osien kestävydestä on erittäin tärkeää tietoa.

Ohjelmalla on jatkossa erittäin hyvät mahdollisuudet kehittyä erittäin tehokkaaksi työkaluksi tuotannon kunnossapito-osastolla. Jatkossa työkalun laajentuessa yrityksen muihin tuotantoyksikköihin kertynyttä dataa voitaisiin vertailla eri yksiköiden välillä. Eri yksiköiden työskentelytavat eroavat toisistaan ja näitä yhdistämällä voitaisiin poimia yksiköiden parhaat toimintatavat.

7.3.1 Väriyksen hyödyntäminen käyttöliittymässä

Testilaitteiden taustaväreissä oli ohjelmistossa aluksi virhe josta johtuen värit eivät näyttäneet oikeaa informaatiota. Näihin saatiin korjaus ja värit näkyvät oikein. Kyse oli tällöin ohjelmiston virheestä. Huollon lähestyessä testilaitte siirtyy ensin keltaiseen tilaan ja huoltorajan ylittyään vaihtaa tilansa punaiseksi. Normaalityltilassa testilaitte näyttää vihreää. Myös värien olisi hyvä näkyä testilaitteen juuressa, mistä huollon tarpeessa olevat testilaitteet olisi helpompi paikallistaa heti ohjelman avatessaan. Kuvassa 16 on koontasolun testilaitteiden värit. TB1-DCT4 on käytössä oleva testimekaniikka ja VICTOR_FINUI on käytössä oleva adapteri.



Kuva 16. Koontasolun näkymän värit.

7.3.2 Raportit

Raporttien vajavaisuus oli ehkä suurimpia ongelmia koko ohjelman käyttöönotossa. Näiden on oltava riittävän monipuolisia, että niistä saadaan kaikki tarvittava tieto. Data on myös saatava jaotelluksi halutun ajan jakson mukaan ja näin ollen aikamäärittelykohta olisi oltava jokaisessa raportin kohdassa. Myös raportointisovellukseen olisi saatava raporttien tarkastelumahdollisuus, koska raportointisovellus on tulevaisuudessa huoltomiesten ainoa työkalu. Vian analysoinnissa raportit ovat hyödyllisiä apuvälineitä. Käyttäjätunnuksen yhteyteen vuorotunnuksen lisääminen olisi myös hyvä apu ajatellen työnjohtajaa, joka seuraa huoltoja ja ohjelman käyttöä omassa vuorossa. Näin raportteja tarkasteltaessa voisi huoltohistorian lajitella oman vuoronsa mukaan.

7.3.3 Raportointisovelluksen historiatiedot

Raportointisovelluksessa ei testilaitteisiin ole saatavilla huoltohistoriaa. Jokaiseen testilaitteeseen on saatava huoltohistoria näkyviin. Esimerkiksi 5 – 10 edellistä huoltotapahtumaa helpottaisi vian paikallistamista ja kannustaisi aktiivisempaan työkalun käyttöön. Tämä säästäisi varmasti myös varaosakustannuksissa ja vähentäisi laitteen seisakkiaikoja.

7.3.4 Laskuri linjan sekä solun juureen

Jos käyttöliittymässä olisi laskuri testilaitteen tai linjan juuressa, tämä auttaisi kehittämään sekä laajentamaan työkalun käyttöä muihin ennakkohuoltokohteisiin. Esimerkiksi koneladonnan laitteet voitaisiin huoltaa tämän laskurin mukaan josta saataisiin linjalta tulleiden puhelimien kappalemäärät. Koneladonnan huollot voitaisiin siirtää myös kappaleiden mukaan tehtäväksi aikaperusteisten huoltojen sijaan. Tämä olisi siinä suhteessa järkevämpää, mikäli linja ei ole jatkuvassa käytössä. Tämä kertoisi valmistuneiden kappaleiden mukaan linjan todellisen huoltotarpeen mitä välttämättä aikaperusteisesti tehtävä huolto ei kerro. Laskurin lisääminen mahdollistaisi myös muiden kuin testilaitteiden siirtämisen MMT-työkaluun. Esimerkiksi ennakkohuollon piiriin kuuluvat ruuvausautomaatit, ruuvaimet ja koontajigit voitaisiin huoltaa kappalemääriin perustuen. Tämä parantaisi laitteiden luotettavuutta ja vastavasti vähentäisi seisakkiaikoja.

7.3.5 Ohjelman vastuhenkilö

Ohjelmalle on määritettävä vastuhenkilö joka vastaa jatkossa ohjelmasta ja kehittämisestä. Jos ohjelma jätetään monen henkilön vastuulle tarkoittaa se useasti, että jaettu vastuu on yhtä kuin ei kenenkään vastuu. Jo käyttöönottoprojektin aikana oli havaittavissa, että tehtäviä joita oli määritelty monen ihmisen vastuulle jäivät ne helposti tekemättä. Vastuut on kohdistettava selkeästi tai niille on määriteltävä vastuhenkilö, jotta ne tulevat tehtyä. Ohjelmalla on tulevaisuudessa paljon mahdollisuuksia ja sitä aktiivisesti kehittämällä on erittäin käyttökelpoinen sovellus. Ohjelmiston tehokas hyödyntäminen säästää huoltohenkilöstön työaikaa ja tehostaa siten huoltotoimintaa.

7.3.6 Käyttöseuranta

Kuten lähes aina kaiken uuden kohtaaminen synnyttää muutosvastarintaa ja tämä havaittiin myös tämän projektin edetessä. Ohjelman käyttöönoton yhteydessä huomattiin, että ohjelmasta puuttuu paljon raportointeja. Tämän seurauksena myös työnjohdolle asennettiin sovellukset, että hekin voivat seurata oman vuoronsa tekemisiä ja antaa näin suoraa palautetta käytöstä. Jatkossa mahdollisesti pohdinnan arvoinen asia olisi työkalun käytön siirtämisen osaksi huoltomiesten bonusjärjestelmää. Lisäksi työkalun kehitystä mietittäessä olisi panostettava riittävästi ominaisuuksiin mitkä avustavat huoltomiesten päivittäistä työtä. Näin työkalu muodostuisi tarpeelliseksi työvälineeksi päivittäisissä huoltotilanteissa.

Tämän hetken tilanteen mukaan koontahenkilö kutsuu huoltomiehen paikalle sytyttämällä sinisen lampun koontasolun signaalimajakkaan. Lampun sytyttäminen voisi käynnistää jatkossa solun juuressa olevan huollon itsestään ja huollon suoritettuaan huoltomies kuittaisi suoritettua toimenpiteen. Näin aikaleima saataisiin alkamaan heti vikatilanteen synnyttyä ja vikatilanteen kesto olisi realistisempi. Tämän ansiosta myös raportoinnin oikeellisuus parantuisi huomattavasti, koska jokaisesta huoltokutsusta jäisi aikaleima työkaluun. Aikaleimoja ja huoltokuittauksia vertailemalla nähtäisiin kuinka paljon raportointeja puuttuu työkalusta.

7.3.7 Testauslaitteiden kalibroinnit

Testauslaitteet olisi syytä kalibroida säännöllisesti. Ohjelmaan voitaisiin lisätä myös kalibrointien tarkastukset jatkossa. Tietyn kappalemäärän saavuttaessa testilaitteesta

tarkistettaisiin teho-, audiotaso ja valonlähteiden tasot. Tämä vähentäisi testauksessa syntyviä testivirheitä ja vähentäisi myös seisakkaikoja vähentämällä vikatilanteiden syntyä. Mikäli käytössä on tilastollinen kalibrointi niin kappalemäärän ylittyessä kaikki vastaavat testilaitteet on syytä kalibroida, koska tietoa etsiessä on samalla helppo hakea tieto kaikista vastaavan tuotteen testilaitteista.

7.3.8 Dokumentit

Ohjelmaan voitaisiin jatkossa miettiä mahdollisuutta dokumenttien tallentamiseen. Esimerkiksi huolto-ohjeet, kytkentäkaaviot ja mahdolliset muut tuotekohtaiset ohjeistukset. Tällä hetkellä ohjeistukset ja huolto-ohjeet löytyvät intranetistä ja yrityksen verkkoasemilta. Tämä mahdollisuus nopeuttaisi ja helpottaisi tiedonhakua mikäli ohjeistukset löytyisivät samasta työkalusta. Dokumenttiosion lisääminen olisi työkaluun suhteellisen helppo toimenpide. Testilaitteelle voisi jatkossa mahdollisesti lisätä teksikentän mihin olisi mahdollista kirjoittaa testilaitetta koskevia viestejä ajatellen muita vuoroja. Esimerkiksi jos testilaitteessa on ollut vikaa, joka ei varmuudella ole tullut kuntoon voitaisiin jättää esimerkiksi viesti ”testilaitte seurannassa audiovian vuoksi”.

8 Yhteenveto

Tämä opinnäytetyö toteutettiin käyttöönottoprojektina, jossa yritykselle otettiin käyttöön työkalu huoltojen seurantaan, hallinnointiin ja analysointiin. Opinnäytetyön tekemisessä avuksi käytettiin yrityksen sisäisiä dokumentteja, yrityksen asiantuntijoiden lausuntoja ja kirjallisuuslähteitä. Projektin aikana tutustuttiin yritykselle aiemmin tehtyihin opinnäytetöihin, jotka koskettavat huoltotoimintaa.[3] [4] [11] Opinnäytetyön tekemisen aikana ohjelmisto otettiin käyttöön tehtaan Flali- ja Finui-alueella. Projektin aikana ylläpidettävät tuotteet konfiguroitiin, ohjelmistoa testattiin ja kehitettiin, osien vaihtorajat määriteltiin, käyttöohjeet päivitettiin, luotiin toimintamalli ja osa aikaperusteisista huolloista siirrettiin eri työkalusta MMT-työkaluun muuttamalla ne kappalemääräperusteisiksi.

Seurantajärjestelmän käyttäjät olivat pääasiassa tyytyväisiä itse ohjelmistoon. Käytön aikana saaduissa palautteissa ohjelmistoon toivottiin työtä helpottavia ja nopeuttavia asioita. Projektin edetessä havaittiin myös paljon uusia soveltamiskohteita, joissa työkalua voitaisiin käyttää. Esimerkiksi tuotannossa olevat automaattiruuvaimet, koontajigit ja ruuvimaljat voitaisiin huoltaa perustuen kappalemääriin.

Kun järjestelmä on ollut käytössä tarpeeksi kauan, aletaan saada todenmukaista tietoa laitteiden ja osien kestävydestä. Tällöin osien huoltorajaa voidaan säätää vastaamaan paremmin todellisuutta. Myös uusia osia saattaa päätyä ennakkohuollon piiriin, koska kaikki huoltotapahtumat kirjataan työkaluun. Työssä havaittiin varosien kestossa olevan todella iso hajonta ja osien ennakkohuoltorajat ovat tämän vuoksi erittäin hankala määritellä. Laskureita seuraamalla voidaan havaita myös esimerkiksi väärästä testauslaitteen säädöistä johtuvia osan vaihtoja vertailemalla eri testilaitteita keskenään.

LÄHTEET

- [1] Rivinoja, J., *Testausalueen ennakkohuoltolaskurin ohjelmointi*, insinööriyö, Turun ammattikorkeakoulu, 2004, 34s.
- [2] Vaartio, J., *Testausalueen ennakkohuoltolaskurin suunnittelu*, insinööriyö, Turun ammattikorkeakoulu, 2004, 40s.
- [3] Lehti, M., *Matkapuhelintuotannon testauksen kunnossapidon ja ennakkohuoltojen kehittäminen*, insinööriyö, Turun ammattikorkeakoulu, 2006, 42s.
- [4] [www-dokumentti]. Opetushallitus, *Kunnossapidon käsitteet ja määritelmät* Saatavilla: http://www.edu.fi/oppimateriaalit/kunnossapito/perusteet_2-1_kunnossapidon_kasitteet_ja_maaritelm.html (Luettu 22.11.2009).
- [5] PSK 6201, Kunnossapito, Käsitteet ja määritteet, PSK Standardisointiyhdistys ry 2003.
- [6] SFS-EN 13306 Standardi, kunnossapitosanasto, Suomen Standarditoimistoliitto SFS ry, 2001.
- [7] Anttonen, K., *Tehosta projektityötä*, TALENTUM 2003, 270s.
- [8] Forsberg, K., Mooz, H., Cotterman, H., *Projektin hallinta*, Gummerrus kirjapaino Oy, Jyväskylä 2003, 350s.
- [9] Bergun, S. *Projektinhallinnan taito*, Gummerrus kirjapaino Oy, Jyväskylä 2006, 476s.
- [10] Numminen, M., *Testausalueen laitteistonhallinta*, insinööriyö, Turun ammattikorkeakoulu, 2003, 83s.
- [11] Lapinleimu, I., Kauppinen, V., Torvinen, S., *Kone- ja metalliteollisuuden tuotantojärjestelmät*, WSOY, 1997, 398s.
- [12] [www-dokumentti]. Kunnossapitoyhdistys Promaint Ry, *Tietopankki* Saatavilla: <http://www.promaint.net/tietopankki> (Luettu 18.11.2009)
- [13] [www-dokumentti]. Yleisen kielitieteen laitos: Helsingin yliopisto *Projektinhallinta* Saatavilla: <http://www.ling.helsinki.fi/kit/2006k/cit310pro/suunnittelu/aikataulus.shtml>. (Luettu 11.11.2009)
- [14] *ABB Oy: TTT-käsikirja*. 2000 Saatavissa [www-muodossa: http://www02.abb.com/global/fiabb/fiabb255.nsf/viewunid/C46D5509D325D21AC225695B002FB07B/\\$file/230_0007.pdf](http://www02.abb.com/global/fiabb/fiabb255.nsf/viewunid/C46D5509D325D21AC225695B002FB07B/$file/230_0007.pdf)
- [15] Jaakkola, V-P., Viinikkala, M., Kuikka, S., *Value Added Web Services For Industrial operations And Maintenance*.2006 5 s.