



ETS Control palopeltien ohjaus- ja valvontajärjestelmän suunnittelutyökalun kehittäminen

Antti Pekkarila

OPINNÄYTETYÖ
Huhtikuu 2019

Talotekniikka
Sähköinen talotekniikka

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Talotekniikka
Sähköinen talotekniikka

PEKKARILA, ANTTI:

ETS Control palopeltien ohjaus- ja valvontajärjestelmän suunnittelutyökalun kehittäminen

Opinnäytetyö 35 sivua, joista liitteitä 5 sivua
Huhtikuu 2019

Opinnäytetyön tehtiin tamperelaisessa Bithouse Oy:ssä. Yritys on erikoistunut kiinteistön ja teollisuusautomaation laite- ja ohjelmistokehitykseen. Työn tavoitteena oli kehittää ETS Control palopeltien ohjaus- ja valvontajärjestelmään uusi suunnittelutyökalu.

Opinnäytetyön suunnitteluvaiheessa käytettiin Excel-taulukkolaskentaa. Suunnitelmat tehtiin manuaalisesti aiemmin käytössä olleisiin Excel-taulukoihin. Kehitystyötä ajatellen yrityksessä päätettiin, että suunnittelua jatkettaisiin edelleen Excel-taulukoiden pohjalta, kuitenkin siten, että hyödynnetään Excel-ohjelmointia, eli käytetään apuna Microsoft Visual Basic for Applications -ohjelmaa. Tällä tavalla automatisoimalla Excel-taulukoiden käyttöä voidaan nopeuttaa suunnittelua ja vähentää suunnittelun aikaisia virheitä.

Työn aluksi suunniteltiin tarkasti käytettävät Excel-taulukot, suunnittelutyökalun toteutusta varten. Tarkoituksena oli, että taulukot sisältävät kaikki suunnittelussa tarvittavat tiedot ja varmistetaan, että suunnittelun lopputulos on halutunlainen. Taulukoiden käytön periaatteena oli, että lähtötiedot kootaan yhteen taulukkoon tietokannan tapaan. Näistä tiedoista prosessoitiin järjestelmän valmis kytkentäkuva.

Työtä toteutettaessa täytyi perehtyä Excel-taulukkolaskentaan syvällisemmin ja varsinkin Excel VBA -ohjelmointiin. Haastavinta työssä oli ohjelmoinnin opettelu. Lopputuloksena saatiin aikaan toimiva suunnittelutyökalu, jota yrityksessä voidaan tulevaisuudessa hyödyntää järjestelmän suunnittelussa ja toteutuksessa.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in Building Services Engineering
Electrical building services

PEKKARILA, ANTTI:
Development tool for ETS Control fire damper system

Bachelor's thesis 35 pages, appendices 5 pages
April 2019

This thesis was made in Bithouse Oy enterprise that is specialized in hardware and software development for building and industrial automation systems. The aim of thesis was to implement a design tool for ETS Control fire damper system.

Originally designing was done by using Excel spreadsheets that were maintained manually. This was arduous, slow and subject to mistakes.

It was decided to plan a new design tool that was still based on Excel spreadsheets but makes use of programming by using Excel Visual Basic for Applications. First it was designed detailed Excel spreadsheet that contain all the necessary data that is used in the design. In this way it was created a kind of database. This way the use of Excel spreadsheets can be automatized. This quickens the design and decreases errors in the documents.

As a result of this thesis it was accomplished a new design tool. It produces all the necessary design documents as well as the diagram of the connection for the input/output combination module.

SISÄLLYS

1	JOHDANTO	6
2	PALOPELLIT	7
	2.1 Palopeltien ohjaus ja valvonta.....	7
	2.2 Palopeltien erot	8
	2.2.1 Mekaaninen palopelti.....	8
	2.2.2 Sähköisellä toimilaitteella oleva palopelti.....	9
	2.3 Lain määräykset palopelleille	11
3	OHJAUS- JA VALVONTAJÄRJESTELMÄ.....	13
	3.1 ETS-Control palopeltien ohjaus- ja valvontajärjestelmä	13
	3.1.1 Järjestelmän rakenne ja käyttö	13
	3.1.2 Magio816 -ohjainkortti	14
	3.1.3 Järjestelmän konfigurointi.....	16
	3.1.4 Järjestelmän liittäminen muuhun RAU-järjestelmään	17
	3.2 Järjestelmän vahvuudet	18
	3.3 Järjestelmän heikkoudet ja ongelmat	19
4	SUUNNITTELUTYÖKALUN KEHITTÄMINEN	20
	4.1 Suunnittelun lähtökohta ja nykytilanne	20
	4.2 Työkalulle asetettavat tavoitteet	21
	4.3 Vaihtoehtoisia toteuttamistapoja	22
	4.4 Suunnittelutyökalun toteutus	22
	4.4.1 Excel VBA ohjelmointi	22
	4.4.2 Suunnittelutyökalun toteuttaminen Excel VBA:lla	24
	4.4.3 Saadut edut työkalusta	31
5	POHDINTA	32
	LÄHTEET	34
	LIITTEET	36
	Liite 1. Moduulien lukumäärän määrääminen	36
	Liite 2. Kaksoisarvojen hakeminen	37
	Liite 3. Tietojen siirtäminen KytKentäkuvat-taulukkoon	38
	Liite 4. Kaaviokuva työkalun toimintaperiaatteesta	40

LYHENTEET JA TERMIT

CPU	Suoritin
DI	Digital input
DO	Digital output
I/O-piste	Tulo/lähtö-liitäntäpiste
Indikoida	Ilmaista, osoittaa
Konfiguroida	Asettaa, määrittää
PP	Palopelti
RAU	Rakennusautomaatio
VBA	Visual Basic for Applications

1 JOHDANTO

Tämän työn tarkoituksena on kehittää ETS -Control palopeltien ohjaus- ja valvontajärjestelmän suunnitteluprosessia. Teen Bithouse Oy:ssä työkseni kytkentäkuvia sekä palopeltijärjestelmien konfigurointia. Opinnäytetyön aihe tulikin työni kautta. Tehdessäni suunnittelua palopeltijärjestelmälle niin huolimattomuusvirheitä tulee aina väkisin prosessin aikana. Eräs tavoite työssäni onkin kehittää sellainen suunnittelutyökalu, joka eliminoisi virheitä ja nopeuttaisi suunnittelua.

Bithouse Oy on vuonna 2015 perustettu muutaman henkilön IT-alan yritys, joka erikoistunut kiinteistö- teollisuusautomaation laite- ja ohjelmistokehitykseen ja muutenkin toimii rakennusautomaation parissa. Yritys on alun perin sijainnut Pirkkalassa, mutta tällä hetkellä yrityksen toimisto sijaitsee Tampereella. Yritys kehittää WEB-pohjaisia automaatiotuotteita ja järjestelmiä asiakkaiden tarpeiden mukaan (Bithouse Oy, 2019).

Bithouse Oy:n myös suunnittelee ja tekee käyttöönoton ETS Nordin valmistamaan palopeltien valvonta- ja ohjausjärjestelmään. Opinnäytetyössäni esittelen järjestelmän toimintaperiaatteen ja laitteistot sekä pyrin kehittämään tähän liittyvää suunnittelutyötä.

2 PALOPELLIT

2.1 Palopeltien ohjaus ja valvonta

Nykypäivän rakennusmääräykset ovat lisänneet tarvetta palonrajoittimien käyttöön, mm. toimilaitteellisten palorajoittimien testausvaatimus on lisääntynyt. Palorajoittimia voidaan kutsua paremmalta nimeltään palopelleiksi (Sähkötieto ry, 2014).

Palo-osastojen välisen palon leviäminen ilmanvaihtokanavia pitkin estetään palopeltien avulla, joiden täytyy kestää määrätyn palonkestoajan. Palopellin toiminta perustuukin siihen, että siinä on erillinen lämpösulake. Sulakkeen lauetessa jousivoimalla toimiva pelti sulkeutuu ja estää palon leviämisen palo-osastosta toiseen. Palosulakkeen laukeamislämpötila on pääsääntöisesti $+70^{\circ}\text{C}\pm 5^{\circ}\text{C}$. Sulakkeen laukeamislämpötila voi vaativissa tiloissa olla alhaisempi henkilöturvallisuuden takia. Joissakin erityistapauksissa sulakkeen laukeamislämpötila voi olla korkeampi, jolloin siitä ei saa aiheutua henkilövaaraa. Tällaisia kohteita voivat olla esim. öljypoltinlaitokset, jossa lämpötilat ovat muutenkin korkeat (Sähkötieto ry, 2014).

Nykypäivänä toimilaitteelliset palopellit ovat yleistyneet. Näissä itse peltiä ohjataan/viritetään erillisellä sähköisellä toimilaitteella. Kuten perinteisessä mekaanisessa palopellissä on palosulake, niin sähköisessä toimilaitteessa on erillinen lämpötila-anturi ilmanvaihtokanavassa. Anturin havaitessa laukeamislämpötilan niin se sulkee palopellin toimilaitteen jousivoimalla. Toimilaitteella varustetut palopellit voidaan myös laukaista ulkoisella laitteella, kuten paloilmoitinjärjestelmällä, savuilmallisilla tai esim. rakennusautomaatiojärjestelmällä (Sähkötieto ry, 2014).

Palopeltien valvonta voidaan jakaa kahteen eri hälytysprioriteettiin. Prioriteetti A:n palopellit sijaitsevat kiinteistössä ilmastointikoneen runkokanavassa ja prioriteetti B:n palopellit ovat ilmastointikoneen haarakanavissa. Näiden kahden hälytysprioriteetin ero onkin siinä, kun palopellin A mennessä kiinni niin se samalla sammuttaa myös ilmanvaihtokoneen. Ilmanvaihtokone sammutetaan

prioriteetti A:ssa sen takia, ettei ilmanvaihtokanavissa alkaisi kanavapaineet nousemaan ja laskemaan, niin että siitä koituisi haittaa. Prioriteetti B:n palopellin mennessä kiinni ilmanvaihtokonetta ei tarvitse välttämättä sammuttaa, koska kanavapaineet eivät muutu niin konkreettisesti, että ilmanvaihtokone täytyisi sammuttaa. (ETS Nord Oy, 2019b)

Palopelti on sellainen, että sen rakenne voi muuttua itse palon aikana ja silloin ne eivät enää toimi. Niillä on yksi turva-asento ja se on kiinniasento. Kun taas verrattaessa palopeltiä savunhallintapelteihin, niin savunhallintapellin täytyy toimia palon aikana, joten sen rakenne ei saa kärsiä suurista lämpötiloista. Joissakin erityiskohteissa pystytään käyttämään laitteita, joissa on palopeltien ja savunhallintapeltien yhdistelmälaitteita. Jos edellä mainittuja peltejä asennetaan kiinteistöön, niille täytyy tehdä riskienarviointi, joka voi johtaa siihen, että palopellit sekä niiden ohjausjärjestelmä luokitellaan sen jälkeen turvajärjestelmäksi (Sähkötieto ry, 2014).

2.2 Palopeltien erot

Palopeltejä on pääasiasiassa kahdenlaisia: mekaanisia palopeltejä ja erillisellä sähköisellä toimilaitteella olevia palopeltejä. Mekaanista palopeltiä ei pysty erikseen ohjaamaan, vaan se pitää virittää käsin. Sähköisellä toimilaitteella varustettu palopelti taas pystytään virittämään palopeltijärjestelmän kautta, ilman että täytyy fyysisesti käydä pellin luona. Sen lisäksi, että on toiminnallisesti kahdenlaisia palopeltejä, niin ne voidaan luokitella eri paloluokkiin (ETS Nord Oy, 2019a).

2.2.1 Mekaaninen palopelti

Perinteinen palopelti on mekaaninen (kuva 1) ja sen toiminta perustuu täysin siinä olevaan palosulakkeeseen. Kun palosulake saavuttaa laukeamislämpötilansa niin se sulkee palopellin sen jousivoimallaan. Mekaanista palopeltiä ei pysty erikseen ohjaamaan mistään muualta kuin fyysisesti palopellin luota. Mekaanisia palopeltejä on kahdenlaisia. Toinen tyyppi on täysin itsenäinen laite, eli siitä ei tiedetä milloin se on auki ja milloin kiinni. Toisen tyyppin palopelti ilmaisee, milloin se on auki tai kiinni, sillä se on varustettu rajakytkimillä, jotka kertovat sen tilan.

Rajakytkimiä on kaksi, joista toinen ilmaisee, milloin palopelti on kiinni ja toinen, milloin se on auki (ETS Nord, 2019a).



KUVA 1. Mekaaninen palopelti (ETS NORD Oy, 2019a)

2.2.2 Sähköisellä toimilaitteella oleva palopelti

Erillisellä sähköisellä toimilaitteella varustetun palopellin (kuva 2) toiminta sen sijaan perustuu sähköiseen ohjaukseen. Toimilaitteen toimintasuunta on sellainen, että kun toimilaitteelle ei syötetä ohjaussähköä, niin se on kiinni. Kun toimilaitteelle syötetään sähkö niin se ohjautuu auki asentoon. Toimilaitte toimii yleensä joko 230 V vaihtojännitteellä tai 24 V tasajännitteellä. Peltimoottori toimii siten, että sille syötettävä ohjaussähkö kulkee kanavassa sijaitsevan lämpöanturin (kuva 3) kautta itse moottoriin. Tämä on tehty turvallisuuden varmistuksen takia, koska kun lämpöanturin havaitessa liian suuren lämpötilan ilmastointikanavassa, niin sen sisällä oleva kosketin aukeaa, joka katkaisee virtapiirin ja minkä seurauksena palopelti menee kiinni (ETS Nord Oy, 2019a).

Sähköisellä toimilaitteella olevaa palopeltiä pystytään lisäksi ohjaamaan erilaisilla ulkopuolisilla laitteilla, kuten esimerkiksi huoneistossa sijaitsevalla palo/savuilmaisimella tai kiinteistön rakennusautomaatiojärjestelmällä. Etu toimilaitteellisessa palopellissä on se, että sitä ei tarvitse käydä fyysisesti paikan päällä virittämässä, vaan se pystytään ohjaamaan auki ja kiinni asentoon yhdestä

paikasta. Tämä tuo myös sen edun, että pelleistä saadaan enemmän tietoa, esimerkiksi niiden historiatietoja, eli milloin palopellit on viimeksi testattu ja onko jokin pelti jumissa (ETS Nord Oy, 2019a).

Sähköisellä toimilaitteella varustetussa palopellissä valvonta tapahtuu samalla tavalla kuin mekaanisessa palopellissä. Siinä on kaksi rajakytkintä auki- ja kiinnitilojen indikointiin. Rajakytkinten kosketintietojen avulla voidaan järjestelmästä nähdä aina, milloin palopelti on joko kiinni tai auki (ETS Nord Oy, 2019a).



KUVA 2. Toimilaitteella ohjattava palopelti (ETS Nord Oy, 2019a)



KUVA 3. Sähköisen palopellin lämpöanturi. (Kuva: Antti Pekkarila 2019)

2.3 Lain määräykset palopelleille

Palopelleille on määritelty monia eri lakeja, standardeja, asetuksia, määräyksiä sekä ohjeita. Palopeltien täytyy läpäistä tyyppihyväksyntä, jotta niitä voidaan asentaa kiinteistöihin ja niiden pitää olla CE-hyväksytyjä. Lisäksi palopellit täytyy testata hyväksytyjen testausmenetelmien mukaisesti. Testauksessa otetaan huomioon palopellin tiiveys mitä kuvataan kirjaimella "E", eristävyys "I" (Sten T, 2019) sekä sen sulkeutuvuus. Palopellin palonkestävyyensaika ilmoitetaan minuutteina merkintöjen E tai EI jälkeen, esimerkiksi EI60 tarkoittaa, että palopelti on tiivis ja eristää lämpöä 60 minuuttia (Ympäristöministeriö, 2017).

Palopellit ovat huollettavia laitteita ja ne täytyy pystyä tarkistamaan ja testaamaan vähintään kuuden kuukauden välein (SFS-EN 15650 2012). Paikat kiinteistössä missä on palopelti, täytyy jättää auki, eli esim. jos palopelti on sijoitettu kattoon alas laskun päälle, niin silloin täytyy tehdä tarvittava huoltoluukku palopeltiä varten. Lisäksi palopellin molemmille puolille ilmastointikanavaan laitetaan puhdistusluukku mahdollista huoltoa varten. Palopelti on sijoitettava ilmastointikanavaan aina, kun kanava lävistää palo-osastoivan rakennusosan. Paikkaan, josta ilmastointikanava lävistää osastoivan rakennusosan, täytyy valita sellainen palopelti, joka täyttää osastoivan rakennusosan paloaikaketovaatimuksen. Palopellit tulee kiinnittää tukevasti osastoivaan

rakennusosaan valmistajan antamia ohjeita noudattaen. Kiinnityksen jälkeen palopelti tulee vielä tiivistää rakennusosaan ja muutenkin hoitaa läpiviennin jälkityöt (Talotekniikkainfo, 2019).

Paloturvallisuusasiat, jotka otettava huomioon rakennushankkeessa, on esitetty Maankäyttö- ja rakennuslain 17. luvun pykälässä 117b. Edellä mainitussa laissa kerrotaan mm. savun ja palon leviämisen rajoittamisesta rakennuksessa. Ilmanvaihdon paloturvallisuuteen liittyvät ohjeet ja määräykset on ennen kerrottu rakennusmääräyskokoelman julkaisuissa E1 ja E7. Rakennusmääräyskokoelman julkaisut E1 ja E7 korvattiin 2017 vuoden lopussa, niin että uudet määräykset ja ohjeet ilmanvaihdon paloturvallisuudesta on esitetty uudessa ympäristöministeriön laissa (Finlex, 2012).

Maankäyttö- ja rakennuslaissa määrätään, että rakennus on suunniteltava ja toteutettava, niin että se on paloturvallinen sen käyttötarkoitukseen edellyttämällä tavalla. Tulipalon mahdollista syttymisvaaraa on rajoitettava. Palon sattuessa rakennuksen kantavien rakenteiden on kestävä vähimmäisajan, ottaen huomioon sen sortuminen, poistumisen turvaaminen, pelastustoiminta sekä palon hallintaan saaminen (Finlex, 2012).

Pelastuslain kolmannen luvun 12. pykälässä kerrotaan laitteiden kunnossapidosta. Edellä mainittu laki toisin sanoen esittää vaatimuksia laitteiden toimintakunnon pitämiseen, huoltoon sekä niiden tarkistukseen. Palorajoittimet ja niiden ohjausjärjestelmät kuuluvat myös pelastuslakiin (Finlex, 2011).

3 OHJAUS- JA VALVONTAJÄRJESTELMÄ

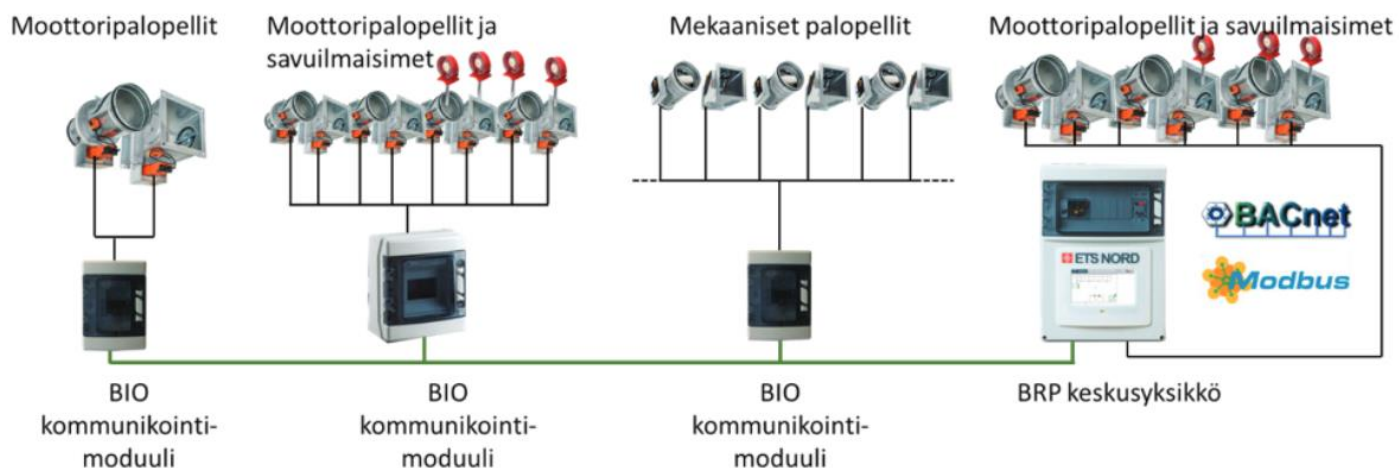
Palopeltien ohjaus- ja valvontajärjestelmällä tarkoitetaan yleensä erillistä järjestelmää, joka liitetään osaksi rakennusautomaatiojärjestelmää. Palopeltien ohjaus- ja valvontajärjestelmän tarkoitus on osittain helpottaa rakennusautomaatiojärjestelmän liiallista paisumista. Tällä tarkoitetaan sitä, että jokainen erillinen järjestelmä pidetään omanaan ja liitetään sitten rakennusautomaatiojärjestelmään. Erillisen palopeltijärjestelmän etu on, että se testaa pellit tietyin väliajoin ja ilmoittaa, jos jokin palopelti on toimi ja vaatii huoltoa. Joissakin tietyissä tapauksissa saatetaan olla hankkimatta erillistä palopeltien ohjaus- ja valvontajärjestelmää, esim. pienissä kiinteistöissä, joissa ei ole montaa palopeltiä. Tällaisessa tapauksessa palopeltien valvonta ja niiden ohjaus hoidetaan rakennusautomaatiojärjestelmästä (Sähkötieto, 2014).

3.1 ETS-Control palopeltien ohjaus- ja valvontajärjestelmä

ETS-Control palopeltien ohjaus- ja valvontajärjestelmä on Bithouse Oy:n ActiWeb käyttöjärjestelmän päälle kehittämä web-pohjainen ohjausjärjestelmä ETS NORD Oy:lle. ETS Control-järjestelmä on yksinomaan kehitetty valvomaa sekä ohjaamaan toimilaitteellisia palopeltejä ja sen suurin etu on sen muunneltavuus. Koska se on hyvin muunneltava, niin se soveltuu hyvin isompiinkin referensseihin, mm Keski-Suomen uusi keskussairaala, joka rakentuu Jyväskylään.

3.1.1 Järjestelmän rakenne ja käyttö

ETS Control -järjestelmä koostuu aina yhdestä keskusyksiköstä, jossa on järjestelmän käyttöpääte (kosketusnäyttö) ja CPU, joka ohjaa koko järjestelmää. Keskusyksikön lisäksi järjestelmä voidaan hajauttaa ympäri kiinteistöä, jolloin täytyy moduulikaapeille kaapeloida Modbus RTU-väyläkaapeli keskusyksiköltä ja 230V sähkönsyöttö kiinteistökeskuksesta (Kuva 4) (ETS Nord Oy, 2019b).



KUVA 4. Esimerkki järjestelmän rakenteesta (ETS Nord Oy, 2019b)

Järjestelmään on mahdollisuus hankkia lisäksi suojattu etäkäyttöohjaus. Silloin keskusyksikköön asennetaan erillinen VPN-modeemi, johon on hankittu operaattorin SIM-kortti. Etäkäyttöohjauksen ansiosta ei aina tarvitse lähteä paikanpäälle ottamaan järjestelmää käyttöön, vaan käyttöönotto voidaan suorittaa etäyhteytenä toimistolta.

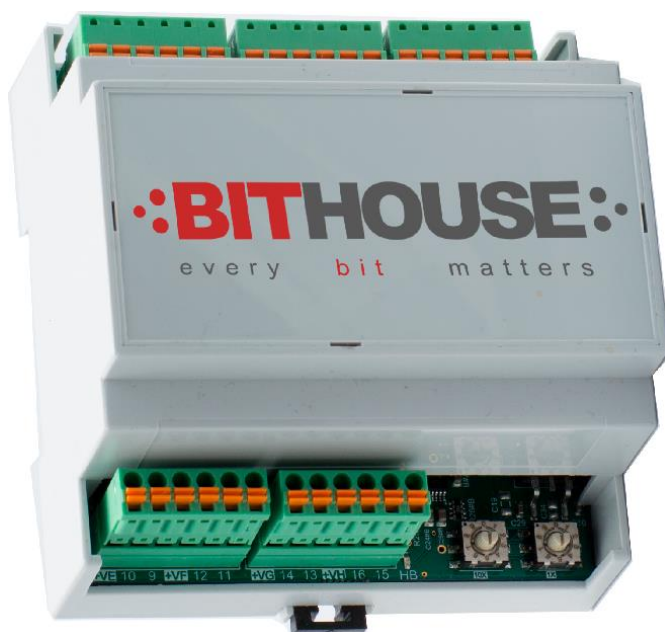
3.1.2 Magio816 -ohjainkortti

Magio816 I/O -moduuli on Bithouse Oy:n kehittänyt ohjainkortti rakennusautomaation erilaisiin sovelluksiin. Moduuli toimii Modbus RTU -väyläkommunikaatiolla, joten se on helppo hajauttaa, jolloin esim. kentälaitteiden kaapelointi helpottuu. Moduulissa on 16 vapaasti ohjelmoitavaa ohjauspistettä, joten siihen on mahdollista kytkeä analogi- ja digitaalilähtöjä sekä -tuloja. Erona analogisissa ja digitaalisissa pisteissä on se, että analogipiste vie kortilta aina kaksi pistettä eli niitä on mahdollista kytkeä vain kahdeksan yhteen moduuliin. Tämä johtuu siitä, että analogipisteet tarvitsevat yhden pisteen nollaa voltia varten ja toisen pisteen itse analogipistettä varten. Digitaalipisteet eivät tarvitse kuin yhden pisteen moduulilta, sillä se ei tarvitse erillistä nollan voltin pistettä. Jos esimerkiksi täytyy liittää neljä analogista pistettä ja viisi digitaalista pistettä, niin silloin riittää yksi moduuli (Taulukko 1) (Monipuolinen Magio816-moduuli, 2018).

TAULUKKO 1. Magio816 -kortin yhteisten pisteiden määrä/kortti (Monipuolinen Magio816-moduuli, 2018)

		Analogisten tulo- ja lähtöviestien määrä									
		0	1	2	3	4	5	6	7	8	
Digitaalisten tulo- ja lähtöviestien määrä	0	■	■	■	■	■	■	■	■	■	■
	1	■	■	■	■	■	■	■	■	■	■
	2	■	■	■	■	■	■	■	■	■	■
	3	■	■	■	■	■	■	■	■	■	■
	4	■	■	■	■	■	■	■	■	■	■
	5	■	■	■	■	■	■	■	■	■	■
	6	■	■	■	■	■	■	■	■	■	■
	7	■	■	■	■	■	■	■	■	■	■
	8	■	■	■	■	■	■	■	■	■	■
	9	■	■	■	■	■	■	■	■	■	■
	10	■	■	■	■	■	■	■	■	■	■
	11	■	■	■	■	■	■	■	■	■	■
	12	■	■	■	■	■	■	■	■	■	■
	13	■	■	■	■	■	■	■	■	■	■
	14	■	■	■	■	■	■	■	■	■	■
	15	■	■	■	■	■	■	■	■	■	■
	16	■	■	■	■	■	■	■	■	■	■

Magio816-moduulia käytetään ETS Control -järjestelmässä palopeltien ohjaukseen. Koska yhteen I/O-korttiin voi kytkeä 16 digitaalista tulo- ja lähtöviestiä, niin siihen teoriassa voi kytkeä viisi toimilaitteellista palopeltiä. Yrityksessä on kuitenkin päätetty, että yhteen Magio816-moduuliin kytetään vain neljä toimilaitteellista palopeltiä. Tällöin saadaan jokainen palopelti kytkettyä omalla irrotettavalle liittimelle (Kuva 5).



KUVA 5. Magio816 I/O-kortti (Bithouse, 2019)

3.1.3 Järjestelmän konfigurointi

Järjestelmän konfigurointi tehdään käyttöönottovaiheessa. Tämä tehdään tässä vaiheessa, koska palopeltien määrä ja kytkentäpaikka on voinut muuttua asennusvaiheessa. Tästä saatava etu on, että järjestelmä vastaa silloin senhetkisiä suunnitelmia ja tehtyjä asennuksia. Toisaalta tämä vaatii sen, että järjestelmän asentajan täytyy olla perillä muuttuneista suunnitelmista sekä asennuksista (Bithouse ActiWeb käyttöönotto ja konfigurointi, 2018).

Konfigurointi alkaa siitä, että vapaasti ohjelmitavaan CPU:hun ladataan valmiiksi ohjelmitu ETS Control palopeltien ohjaus- ja valvontajärjestelmä. Tämä tapahtuu ennen kuin keskusyksikkö lähetään asiakkaalle, sillä CPU asennetaan keskusyksikköön.

Järjestelmän varsinainen konfigurointi on suunniteltu tehtäväksi esim. Excel- taulukkoon, johon listataan kaikki järjestelmässä olevat palopellit (taulukko 2). Taulukkoon merkitään palopeltien positiot. Lisäksi merkitään, onko palopelti varustettu erillisellä toimilaitteella ja mihin palopelti on fyysisesti kytketty (moduuli osoite ja moduulin kanava). Taulukkoon voidaan myös haluttaessa nimetä, missä palopelti sijaitsee kiinteistössä. Tämä tosin vaatii sen, että suunnittelija on merkinnyt palopeltien laiteluetteloon niiden sijainnit (Bithouse ActiWeb käyttöönotto ja konfigurointi, 2018).

TAULUKKO 2. Esimerkki konfiguraatiolistasta (Pekkarila, 2019)

A	B	C	D	dataname	class	group	location	inDeviceTypeFD	inDeviceAddrFD	inChannelFD	outDeviceTypeFD	outDeviceAddrFD	outChannelFD
PPOK17/	S_	PP1_	K1_19	PPOK17/S_PP1_K1_19	motorizedDamper	ppRyhmat/S_HAARAT	Kellari	metz_MR_D110	51	MI1	metz_MR_DO4	49	DO1
PPOK17/	G307TK	PP1_	K1_13	PPOK17/G307TK_PP1_K1_13	motorizedDamper	ppRyhmat/G307_HAARAT	Kellari	metz_MR_D110	51	MI2	metz_MR_DO4	49	DO2
PPOK17/	S_	PP1_	K1_14	PPOK17/S_PP1_K1_14	motorizedDamper	ppRyhmat/S_HAARAT	Kellari	metz_MR_D110	51	MI4	metz_MR_DO4	49	DO3
PPOK17/	G307TK	PP1_	K1_12	PPOK17/G307TK_PP1_K1_12	motorizedDamper	ppRyhmat/G307_HAARAT	Kellari	metz_MR_D110	51	MI5	metz_MR_DO4	49	DO4
PPOK17/	G307TK	PP1_	K1_11	PPOK17/G307TK_PP1_K1_11	motorizedDamper	ppRyhmat/G307_HAARAT	Kellari	metz_MR_D110	52	MI1	metz_MR_DO4	50	DO1

Kun konfiguraatiolista on saatu tehtyä, niin voidaan Excel-taulukko kopioida ja tuoda järjestelmään, jolloin järjestelmän CPU osaa automaattisesti luoda kopioidun listan pohjalta tarvittavat pisteet tietokantaan (kuva 6). Kaikki pisteet voitaisiin luoda käyttöjärjestelmässä ilman, että tehdään mitään erillistä Excel-listaa, mutta siinä menisi turhaan aikaa ja se olisi työlästä.

▼ H305TKPK/

- ▣ PP15_1[schema: fireDamper] (+) (C) (R) (-)
- ▣ PP15_1_BO[schema: BO] (+) (C) (R) (-)
- ▣ PP15_1_BUS_AL[schema: alarm] (+) (C) (R) (-)
- ▣ PP15_1_FUSE_AL[schema: alarm] (+) (C) (R) (-)
- ▣ PP15_1_JAM_AL[schema: alarm] (+) (C) (R) (-)
- ▣ PP15_1_MI[schema: MSI] (+) (C) (R) (-)
- ▣ PP15_1_SW[schema: BI] (+) (C) (R) (-)

KUVA 6. Esimerkki yhden palopellin tietokantapisteistä (Pekkarila, 2019)

Kuvassa 6 olevat palopellin järjestelmäpisteiden nimeäminen tulee BACnet-standardista (Ashrae, 2012). Palopelleilla on vain kolme fyysistä I/O-pistettä, mutta se vaatii järjestelmään seitsemän ohjelmallista pistettä. Esimerkiksi *PP15_1_BUS_AL* tarkoittaa, ettei kommunikaatioväylä saa yhteyttä kyseiseen palopeltiin.

3.1.4 Järjestelmän liittäminen muuhun RAU-järjestelmään

Järjestelmä on mahdollista liittää osaksi kiinteistön rakennusautomaatiojärjestelmää, mutta sen on mahdollista toimia itsenäisestikin. Kun ETS Control -järjestelmä liitetään osaksi rakennusautomaatiojärjestelmää, niin saadaan mm. hälytyksiä palopeltien laukeamisesta tai esimerkiksi ohjaukaskäsky rakennusautomaatiojärjestelmästä (ETS Nord Oy, 2019b)

ETS-Control palopeltien ohjaus- ja valvontajärjestelmä pystytään liittämään RAU-järjestelmään kolmella eri tavalla. Ensimmäinen tapa on viedä hälytykset ja tarvittavat indikoinnit I/O-pisteiden avulla, eli käyttäen ohjaukaskortin DI- ja DO-pisteitä. Edellä mainitun tavan hyvä puoli on siinä, että saadaan varma toiminta järjestelmien välille. I/O-pisteiden käytön ongelma on siinä, haluttaessa lisätä hälytyksiä ja indikoiteja järjestelmien välillä, niin kaapelin pariin määrä kasvaa (ETS Nord Oy, 2019b).

Kaksi muuta tapaa siirtää tietoa RAU-järjestelmän ja ETS Control -järjestelmän välillä, on joko BACnet IP-, Modbus TCP/RTU -kommunikointiväylä. Edut mitä

kommunikointiväylästä saadaan verrattuna I/O-pisteiden käyttöön on se, että tietoa pystytään siirtämään huomattavasti enemmän RAU-järjestelmän ja ETS Control -järjestelmän välillä. Lisäksi saadaan enemmän tietoa siirrettyä järjestelmien välillä. Kommunikointiväylä tarvitsee parikierretystäkaapelista vain yhden parin. Modbus TCP/RTU -kommunikaatioväylän ongelma piilee siinä, kun järjestelmiä konfiguroidaan toimimaan yhdessä, niin tarvitaan kummankin järjestelmän Modbus-rekisterilistat. Rekisterilistojen konfigurointi onkin yleensä hankalaa ja aikaa vievää, jotta kommunikaatioväylä saataisiin toimimaan. BACnet IP -kommunikointiväylän ongelma on se, että Suomessa kyseinen kommunikointiväylä ei ole vielä tarpeeksi tunnettu. Tästä johtuen harvalla RAU-järjestelmätoimittajalla on tarvittavia toimilaitteita, jotta järjestelmät pystyisivät kommunikoimaan keskenään (ETS Nord Oy, 2019b).

3.2 Järjestelmän vahvuudet

Tällä hetkellä ETS Control -järjestelmän vahvuuksiin kuuluu se, että jokaiseen kohteeseen, johon asennetaan kyseinen järjestelmä, teetetään omat moduulikeskukset ja keskusyksikkö. Keskusyksikön ja moduulikaappien teettäminen saattaa voi olla hieman kalliimpaa, verrattuna esim. muihin kilpailijoihin, joilla on oma palopeltien ohjaus- ja valvontajärjestelmä. Investoimalla ETS Control -järjestelmään säästytään asennusaikaisessa kaapeloinnissa ja jos esim. palopeltien määrä tai sijainti muuttuu kiinteistössä, niin järjestelmän päivittäminen vastaamaan uusia suunnitelmia on helppoa.

ETS Control on tällä hetkellä Suomen markkinoilla ainut palopeltien ohjaus- ja valvontajärjestelmä, joka pystyy valvomaan toimilaitteellisen palopellin lisäksi mekaanisia palopeltejä. Lisäksi sen vahvuuksiin kuuluu sen erinomainen seitsemän tuuman kosketusnäytöllinen käyttöpääte, josta pystytään valvomaan ja ohjaamaan kaikkia kiinteistön palopeltejä. ETS Control -järjestelmän vahvuus on siinä, että se käytettävissä on monta eri vaihtoehtoa, miten se liitetään RAU-järjestelmään.

Lisäksi Bithouse Oy:n tekemä ActiWeb-ohjelmiston tuki on taattu hyvin pitkälle tulevaisuuteen, koska ohjelmisto pohjautuu avoimille ohjelmistoille ja ratkaisuille.

Tästä saadaan se hyöty, että jos laitealusta rikkoutuu, niin saadaan aina korvaava laite tilalle (ActiWeb-ohjelmisto, 2019).

3.3 Järjestelmän heikkoudet ja ongelmat

Kuten kaikissa automaatiojärjestelmissä, niin ETS Control -järjestelmässäkin on tiettyjä heikkouksia ja ongelmia. Suurimmat ongelmat tällä hetkellä on järjestelmän suunnittelu, mitä tässä opinnäytetyössä pyritään parantamaan ja kehittämään.

Toinen järjestelmän heikkous on siinä, että jos kiinteistöön tulee paljon palopeltejä niin tämä myös tarkoittaa, että tarvitaan paljon Magio816-ohjainkortteja. Tästä seuraa se ongelma, että Modbus RTU-kommunikointiväylässä on paljon ohjainkortteja, jolloin kommunikointiväylän prosessointiviive muodostuu pitkäksi, mikä hidastaa järjestelmää huomattavasti. Edellä mainittu ongelma ei varsinaisesti ole järjestelmä vika, vaan Modbus RTU-kommunikointiväylän yksi ominaisuus.

Eräs järjestelmän haaste on myös sen konfigurointi. Sillä konfigurointi, kun tehdään esim. Excel-taulukon, niin se on hieman työlästä ja siinä on mahdollisuus tehdä huolimattomuusvirheitä. Tähänkin ongelmaan pyritään hieman paneutumaan tässä opinnäytetyössä.

4 SUUNNITTELUTYÖKALUN KEHITTÄMINEN

4.1 Suunnittelun lähtökohta ja nykytilanne

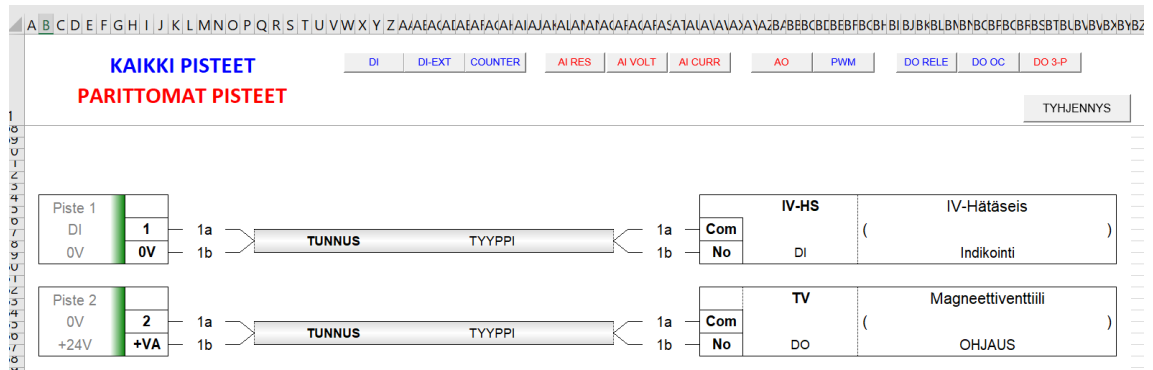
Palopeltijärjestelmän suunnittelu alkaa Bithouse Oy:ssä yleensä siitä, kun ETS Nord Oy on myynyt järjestelmän asiakkaalle, joka yleensä on jonkin rakennustyömaan ilmanvaihtourakoitsija. ETS Nord Oy saa asiakkaalta kaikki mahdolliset dokumentit liittyen palopeltien toteutukseen kiinteistössä. Seuraavassa on lueteltu dokumentit, jotka mahdollistavat hyvän palopeltijärjestelmän suunnittelun:

- Automaatiosäätökaavio, jossa määritetään millä tavalla palopeltijärjestelmä liitetään RAU-järjestelmään.
- Palopeltien kaapelointikaavio, jossa sähkösuunnittelija on määrittänyt kuinka palopeltijärjestelmä tulisi kaapeloida.
- Palopeltiluettelo, jossa on kiinteistön kaikkien palopeltien positiot, mahdollisesti niiden fyysinen koko ja missä ne sijaitsevat kiinteistössä.

Edellä mainituista dokumenteista kaikkia on tosin harvoin suunniteltu kohteeseen. Jokainen suunnittelutoimisto tekee muutenkin suunnitelmat omalla tavallaan, joten jokaisen palopeltijärjestelmän suunnittelukin on aina erilainen. Esim. sähkösuunnittelija on harvoin suunnitellut kaapeloinnin palopeltijärjestelmälle. Tällöin täytyy itse suunnitella, mihin palopeltien moduulikaapit sijoitetaan kiinteistössä ja kuinka moduulikaappien välinen kaapelointi hoidetaan.

Sen jälkeen, kun asiakas on lähettänyt kaikki saatavilla olevat dokumentit liittyen palopeltijärjestelmään, alkaa varsinainen kytkentäkuvien ja moduulikeskusten suunnittelu. Suunnittelu aloitetaan tarkistamalla kaikki dokumentit ja niiden perusteella tehdään järjestelmän kytkentäkuvat ja tarvittaessa kommunikaatioväyläkavio. Yleensä jos on pelkkä palopeltiluettelo saatavilla, niin siinä kerrotaan, missä palopellit sijaitsevat kiinteistössä. Tämän dokumentin perusteella on helppo päättää, että esim. jokaiseen kerrokseen asennetaan yksi moduulikeskus, johon saman kerroksen palopellit kytketään.

ETS Control palopeltien ohjaus- ja valvontajärjestelmän kytkentäkuvat on suunniteltu tällä hetkellä Excel-taulukon avulla. Harri Ollila on koodannut Excel-taulukon Bithousen käyttöön. Tällä on tähän asti suunniteltu kaikki kytkentäkuvat Magio816-ohjainkortille, jota käytetään mm. palopeltien kytkentäkuvien tekemiseen. Kyseistä kytkentäkuvien tekemiseen laadittua Excel-taulukkoa on automatisoitu sen verran, että siinä on komentopainike jokaista korttiin kytkettävää I/O-pistettä kohden (kuva 7).



KUVA 7. Nykyinen Excel-suunnittelutaulukko (Pekkarila, 2019)

Nykyinen Excel-taulukkotyökalu toimii moitteettomasti, mutta sen hankaluus on siinä, että suurin osa suunnittelusta täytyy tehdä käsin taulukkoon. Lisäksi jos projektiin tulee monta Magio816-moduulia, niin jokaisen moduulin kytkentäkuva täytyy tehdä omaan Excel-taulukkoon.

Suunnittelua tulisi parantaa niin, että kytkentäkuvat olisi helppo toteuttaa ja ettei jokaista kytkentämoduulia täytyisi laittaa omaan Excel-taulukkoonsa. Olisi parempi, että kaikki kytkentäkuvat olisivat samassa Excel-taulukossa omilla tulostussivuillaan.

4.2 Työkalulle asetettavat tavoitteet

Uuden suunnittelutyökalun tavoitteena on parantaa nykyisin käytössä olevaa järjestelmää ja korjata siinä olevia puutteita. Suunnittelutyökalun tulisi olla mahdollisimman yksinkertainen käyttää, jotta muutkin yrityksessä työskentelevät voisivat käyttää sitä. Suunnitteluprosessin tulisi edetä johdonmukaisesti, siten että virheitä tulisi mahdollisimman vähän sen aikana. Lopputuloksena pitäisi

saada aikaan käyttökelpoiset ja luotettavat dokumentit. Lisäksi suunnitteluprosessin tulisi nopeutua.

4.3 Vaihtoehtoisia toteuttamistapoja

Kun työkalulle ja lopputulokselle oli määritelty edellä kuvatun mukaiset tavoitteet, aloimme miettiä millä eri tavoilla työkalu voitaisiin toteuttaa. Vaihtoehtoisia menetelmiä voisi olla jokin suunnitteluohjelmisto (esim. AutoCAD, CADS), nykyisen Excel-taulukoiden käyttöön perustuvan työkalun kehittäminen tai kokonaan uusi Excel-työkalu, jossa hyödynnetään VBA-ohjelmointia.

Eri vaihtoehtoja pohtiessamme, päädyimme siihen, että tehdään kokonaan uusi työkalu hyödyntäen Excel VBA-ohjelmointia. Päätös miksi uusi suunnittelutyökalu tehtäisiin Excel VBA:lla oli se, että firmassa oli jo valmiiksi käytössä Excel ohjelmisto, joten ei ollut tarvetta hankkia mitään uutta ohjelmistoa. Tällä tavoin saatiin tehtyä myös säästöä firmassa, kun esim. oltaisiin hankittu jokin suunnitteluohjelmisto, minkä ohjelmistolisenssi voi maksaa useita satoja euroja vuodessa. Toinen ongelma suunnitteluohjelmistossa olisi ollut se, että sen opetteluun olisi kulunut aikaa, jotta tarvittavista dokumenteista saataisiin halutunlaisia. Etu Excel VBA-ohjelmointiin on se, että siihen löytyy todella paljon opetusmateriaalia jo pelkästään netistä, niin ohjelmointi ei tulisi olemaan suuri kynnys, vaikka ei olisikaan ohjelmoinut ennen.

4.4 Suunnittelutyökalun toteutus

Kuten edellä todettiin, päädyimme siihen, että uusi suunnittelutyökalu toteutetaan siis uusien Excel-taulukkojen avulla. Taulukkojen ylläpidossa hyödynnetään Excel VBA -ohjelmointia. Excel VBA:n voidaan automatisoida toimintoja, jotka aikaisemmin täytyi tehdä käsin. Automatisointi nopeuttaa suunnitteluprosessia ja vähentää suunnittelun aikaisia virheitä. Suunnittelutyökalun toteuttaminen edellyttää siis hyvää Excel-ohjelmiston käytön hallintaa ja lisäksi VBA:n käyttöä haluttujen toimintojen ohjelmointiin.

4.4.1 Excel VBA ohjelmointi

Suurin osa ihmisistä, jotka käyttävät Excel-ohjelmaa, jopa päivittäin eivät tiedä kuinka laaja ja monipuolinen se oikeasti onkaan. Visual Basic for Applications -ohjelmalla pystytään Excel-taulukoita automatisoimaan ja ylipäättään laajentamaan Excelin käyttöä huomattavasti. VBA onkin todella laaja aihepiiri ja se vaatiikin useita tunteja, jotta sen oppii hyvin. Excel-taulukossa pystytäänkin nauhoittamaan toimenpidesarjoja, näitä kutsutaan makroiksi. Makrojen nauhoittaminen onkin yksinkertaisin esimerkki ohjelman perehtymiseen. Makrojen nauhoittamisen ja VBA:n yhteys onkin siinä, että kun Excel-taulukossa nauhoitetaan makroja, Excel kirjoittaa VBA-ohjelmakoodia valmiiksi. Mutta paras hyöty saadaan VBA-ohjelmoinnista vasta sitten kun aletaan itse kirjoittamaan ohjelmakoodia. VBA-ohjelmointia pystytään hyödyntämään myös muissa Microsoft Office ohjelmissa, esim. Wordissa ja PowerPointissa (EXCEL VBA-OHJELMOINTI, 2013).

VBA-ohjelmointia aloitettaessa täytyy Excelin valintanauhaa muokata sillä tavalla, että kehitystyökalut tulevat käyttöön. Tämä tapahtuu valitsemalla Excelissä *Tiedosto / Asetukset / Valintanauhan mukauttaminen*, minkä jälkeen valitaan *Kehitystyökalut Päävälilehdet*-listalta ja kuitataan valinnat painamalla *OK* (EXCEL VBA-OHJELMOINTI, 2013).

Ennen kuin ohjelmointia aloitetaan, on Excelin makrosuojauksen tarkistettava. Makrosuojauksen tarkoituksena on estää tai sallia makrojen suorittaminen. Tämä on tarpeen koska makrot voivat sisältää vahingollista koodia, esim. viruksia (EXCEL VBA-OHJELMOINTI, 2013).

Muiden ohjelmointikielten tapaan koodi kirjoitetaan editorilla, joka tässä tapauksessa on Visual Basic -editori. Sen avulla ohjelmakoodi tallentuu ns. moduuleihin. Yhdessä Excel-työkirjassa voi olla useita moduuleita. Nauhoitusta käytettäessä ohjelmakoodi tallentuu omaan moduuliinsa. Jälkikäteen koodit voidaan yhdistää esim. yhteen moduuliin (EXCEL VBA-OHJELMOINTI, 2013).

Editorin käyttö on sikäli helppoa, että Visual Basic ei vaadi kovin tarkkaa kielioppia (syntaksia), kuten esim. C-kieli. Visual Basic tarkistaa ja huomauttaa mahdolliset syntaksivirheet jo ohjelmakoodia kirjoittaessa. Editori tunnistaa myös ohjelmointikielen ns. varatut sanat. Jos esim. kirjoitetaan pienillä kirjaimilla lause:

```
worksheets("Pisteluetello").select
```

niin lause muuttuu muotoon:

```
Worksheets("Pisteluetello").Select
```

eli editori tunnistaa ohjelmointikielen varatut sanat Worksheets ja Select.

VBA -ohjelmoinnissa on oleellista tuntea sen käyttämä objektimalli. Koska suurin osa toiminnoista kohdistuu objekteihin, joita ovat esim. Worksheets (laskentataulukko), Range (solualue). Ohjelmakoodi koostuu pääasiassa objektien valitsemisesta, niihin liittyvien menetelmien käyttämisestä ja objektien ominaisuuksien lukemisesta ja asettamisesta. Esimerkiksi:

```
Worksheets("Aputaulukot").Select
Range("E1:N1").Select
Application.CutCopyMode = False
Selection.Copy
```

Tällä koodilla valitaan Aputaulukot-taulukosta alue alkaen solusta E1 ja päättyen soluun N1, joka alue kopioidaan.

Tärkeintä VBA-ohjelmoinnissa edellä mainittujen seikkojen lisäksi on ymmärtää muuttujien käyttäminen ja niiden tietotyypit, taulukkomuuttujat ja ohjelmarakenteet (esim. toistorakenne).

4.4.2 Suunnittelutyökalun toteuttaminen Excel VBA:lla

Suunnittelujärjestelmän rungon muodostavat samassa Excel-työkirjassa olevat taulukot:

- **Laiteluettelo**, johon määritetään järjestelmään tulevat laitteet ja niiden positiot.
- **Pisteluetello**, joka luodaan laiteluettelon pohjalta ja johon määritetään moduulit ja niihin tulevat liitännät.
- **KytKentäkuvat**, joka luodaan pisteluettelon pohjalta. Tämän taulukon pohjalta pystytään kytkemään laitteet ETS Control -järjestelmään.
- Lisäksi toteutuksessa tarvitaan taulukot **Aputaulukot**, **Tietokanta** ja **Pohja1**.

Edellä selvitetyin mukaisesti aloitin työkalun suunnittelun laatimalla kyseiset taulukot.

Laiteluettelo-taulukon rakenne on kuvan 8 mukainen.

	A	B	C	D	E	F	G
	Tee pisteluettelo						
1							
2	Laitetyyppi		Positio		Tunnus	Kaapelityyppi	Keskus
3	Palopelti	G308	PP17				
4	Palopelti	G309	PP16				
5	Palopelti	G310	PP15				
6	Palopelti	G311	PP14				
7	Palopelti	G312	PP15				

KUVA 8. Kuvakaappaus laiteluettelosta (Pekkarila, 2019)

Laitetyyppi kohtaan määritetään, mikä automaatiolaite on kyseessä. Tähän sarakkeeseen tuleva tieto haetaan Tietokanta-taulukosta (Taulukko 3). Tässä työssä perehdytään pelkästään palopeltien ohjaukseen ja niiden valvontaan. Muut laitetypit voidaan ottaa huomioon tulevissa suunnittelujärjestelmä versioissa.

TAULUKKO 3. Esimerkki tietokantapisteistä

Laitetyyppi
Palopelti
Taajuusmuuttaja
Savuilmaisin (DI)
Lämpötilamittaus
Venttiili
Savuilmaisin (AI)

Laiteluettelo-taulukosta laitetiedot kopioidaan **Pisteluettelo**-taulukkoon. Tämä voitaisiin tehdä helposti kopioimalla tiedot manuaalisesti. Kopiointi on kuitenkin automatisoitu komentopainikkeen **Tee pisteluettelo** avulla. Samalla alustetaan Pisteluettelo-taulukon sarakkeet. Komentopainikkeen ohjelmointi on tässä tapauksessa suhteellisen yksinkertaista. Ennen varsinaista ohjelmointia testasin, miten kopiointi onnistuu nauhoitetun makron avulla. Nauhoitetusta makrosta sain pohjan varsinaisen komentopainikkeen ohjelmointiin. Komentopainikkeeseen **Tee pisteluettelo** ohjelmoidut toiminnot ovat pääpiirteissään seuraavat:

1. Vanhojen Pisteluettelo-taulukossa mahdollisesti olevien tietojen poistaminen.

2. Laiteluettelo-työkalussa olevien sarakkeiden A-C kopiointi Pisteluettelo-työkaluun (kopioidaan vain tiedot).
3. Pisteluettelo-työkalun sarakkeiden E, G, H, J, K, M ja N muotoilun alustaminen siten, että E-sarakkeeseen (Moduulin osoite) voidaan syöttää ohjausjärjestelmästä johtuen vain arvoja 1 ... 99, sarakkeisiin H, K ja N arvoja 1 ... 16 sekä sarakkeisiin G, J ja M arvoja +VA ...+VH. Näitä kelpoisuusmäärittelyitä varten laadin apu-työkalun (työkalu **Apu-työkalu**), josta kelpoisuusvaatimusten mukaiset tiedot haetaan.

Edellä mainittuja toimintoja ohjelmoidessani käytin siis hyväksi Excel VBA:ssa olevaa makron nauhoittamista. Lisäksi pohdin, miten voin kopioida vain tarpeellisen määrän rivejä. Tämän ongelman ratkaisin siten, että käytetään VBA:n For -Next -toistorakennetta (Erilaisia silmukkarakenteita, 2005). Tämä on käyttökelpoinen, koska kopioitavien rivien lukumäärä on tiedossa, sillä kopiointi alkaa työkalun rakenteesta johtuen aina riviltä kolme päättyä työkalun viimeiselle riville. Viimeisen rivin määrittäminen voidaan tehdä monella tavalla. Tässä ohjelmassa käytin lausetta:

```
r1 = Cells(Rows.Count, 1).End(xlUp).Row
```

missä muuttuja r1 saa arvoksi työkalun viimeisen rivin rivinumeron.

Muutenkin ohjelmoinnissa olen käyttänyt runsaasti For-Next -toistorakennetta, koska usein käsitellään työkaluiden tiettyjä rivejä ja sarakkeita. Tämä luonnollisesti edellyttää, että työkaluiden rakennetta ei muuteta, jolloin viitataan aina oikeisiin työkalun soluihin tai solualueisiin. Näin ollen työkaluiden rakenteet täytyy miettiä tarkkaan ennen kuin ohjelmointia aloitetaan.

Pisteluettelo-työkalun rakenne on seuraavan kuvan mukainen (kuva 9).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	Lajittele ja tarkista			Tee kytkentäkuva										
1														
2	Laitte	Positio		Moduuli (osoite)			Pisteet							
3	Palopelti	G308	PP17			DI-EXT		DI-EXT				DO		
4	Palopelti	G309	PP16			DI-EXT		DI-EXT				DO		
5	Palopelti	G310	PP15			DI-EXT		DI-EXT				DO		
6	Palopelti	G311	PP14			DI-EXT		DI-EXT				DO		
7	Palopelti	G312	PP15			DI-EXT		DI-EXT				DO		
8	Palopelti	G313	PP16			DI-EXT		DI-EXT				DO		
9	Palopelti	G314	PP17			DI-EXT		DI-EXT				DO		
10	Palopelti	G315	PP18			DI-EXT		DI-EXT				DO		
11	Palopelti	G316	PP19			DI-EXT		DI-EXT				DO		
12	Palopelti	G317	PP20			DI-EXT		DI-EXT				DO		
13	Palopelti	G318	PP21			DI-EXT		DI-EXT				DO		
14	Palopelti	G319	PP22			DI-EXT		DI-EXT				DO		
15	Palopelti	G320	PP23			DI-EXT		DI-EXT				DO		
16	Palopelti	G321	PP24			DI-EXT		DI-EXT				DO		
17														
18														

KUVA 9. Esimerkki pisteluettelosta (Pekkarila, 2019)

Sarakkeisiin A-C on siis kopioitu vain arvot Laiteluettelo-tilukosta. Tähän taulukkuun tehdään varsinaisen järjestelmän toteutussuunnitelma eli mitä moduuleita ja moduuleiden liitäntöjä tullaan käyttämään. Kelpoisuustarkistuksen vuoksi taulukkuun voi syöttää vain siihen kelpaavia arvoja.

Kun tarvittavat tiedot on syötetty Pisteluettelo-tilukkuun, on tämän jälkeen taulukko

- lajiteltava (lajitteluperusteena sarakkeissa E ja H olevat arvot)
- tarkistettava, että saman moduulin sisällä ei ole käytetty samoja liittinumeroita.

Edellä mainitut toiminnot tehdään komentopainikkeeseen **Lajittele ja tarkista** liitetyn koodin avulla. Lajittelun ohjelmointi on suhteellisen yksinkertaista, koska siinä voidaan hyödyntää Excelissä valmiina olevaa toimintoa *Tiedot/Lajittele*. Nauhoitin lajittelumakron, jossa lajittelu tapahtuu sarakkeissa E ja H olevien arvojen perusteella. Näin samaan moduuliin liittyvät rivit saadaan allekkain moduulinumeroiden mukaiseen järjestykseen pienimmästä suurimpaan.

Tietojen tarkistamien on hieman laajempi kokonaisuus. Lähdin ratkaisemaan ongelmaa siten, että mietin millaisella koodilla voisin ensin selvittää, montako erilaista moduulia Pisteluettelo-tilukko sisältää. Ajattelin, että käyttökelpoinen ratkaisu olisi hyödyntää ohjelmoinnissa taulukkomuuttujia. Koska moduuleita maksimissaan voi olla 99, otin käyttöön taulukkomuuttujan nimeltä moduli(), joka ohjelmakoodissa määritellään seuraavasti:

```
Dim moduli(100) As Integer
```

Tämä tarkoittaa, että moduli()-taulukko voi sisältää 100 alkiota, joiden tietotyypinä on kokonaisluku. Itse ohjelmakoodi on toteutettu tässäkin For-Next-toistorakennetta käyttämällä. Koodin periaate on se, että käydään läpi kaikki Pisteluettelo-taulukon rivit ja määrätään montako erilaista moduuliosoitetta on määritelty. Näin ollen moduli(1) saa arvoksi Pisteluettelo-taulukon ensimmäisellä/ensimmäisillä rivillä/riveillä olevan arvon. Ohjelmakoodi on liitteessä 1.

Seuraavaksi täytyy ohjelman avulla selvittää, montako riviä kukin moduuli käsittää (maksimissaan rivejä voi olla neljä ohjausjärjestelmän rakenteesta johtuen). Tämäkin toiminto voidaan toteuttaa For - Next-toistorakenteen avulla. Koodissa käytetään laskurimuuttujaa, jonka arvot sijoitetaan taulukkomuuttujaan rivit (). Jos esimerkiksi rivit (2) = 3, niin tämä tarkoittaa, että Pisteluettelo-taulukossa moduulisoihteessa kaksi on kolme riviä.

Taulukoiden moduli () ja rivit () avulla voidaan tarkistaa Pisteluettelo-taulukossa mahdollisesti olevat kaksoisarvot (engl. duplicate). Edellä mainittuja taulukoita hyödynnetään myös myöhemmin, kun muodostetaan KytKentäkuvat-taulukko, jossa jokaiselle moduulille laaditaan oma tulostettava sivu, josta selviää moduuliin tulevat liitännät. Kaksoisarvojen hakemisessa periaate on, että jokaisen moduulin sisällä tarkistetaan mahdolliset kaksoisarvot siten, että kerrallaan jokaista arvoa verrataan kaikkiin muihin moduulin liitäntöjen arvoihin. Ratkaisin ongelman siten, että Pisteluettelo-taulukossa olevat arvot (sarakkeet H, K ja N) luetaan kolmiulotteiseen taulukkoon arvot (k, i, j), missä k on moduulin (järjestys)numero, i on moduuliin liittyvän rivin numero (1 ... 4) ja j on sarakenumero (1 ... 3). Koodi kokonaisuudessaan liitteessä 2.

Jos Pisteluettelo-taulukko on esimerkiksi kuvan 10 mukainen, niin esimerkiksi arvot (3,2,3) saa arvon 7 (3 → moduuli 4, rivi 2, sarake 3).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	Lajittele ja tarkista			Tee kytkentäkuva										
1														
2	Laite	Positio	Moduuli (osoite)	Pisteet										
3	Palopelti	G308	PP17	2	DI-EXT	+VE	3	DI-EXT	+VH	11	DO	+VE	7	
4	Palopelti	G309	PP16	2	DI-EXT	+VB	8	DI-EXT	+VG	4	DO	+VG	8	
5	Palopelti	G310	PP15	3	DI-EXT	+VE	1	DI-EXT	+VB	13	DO	+VF	11	
6	Palopelti	G311	PP14	3	DI-EXT	+VD	3	DI-EXT	+VA	3	DO	+VD	5	
7	Palopelti	G312	PP15	4	DI-EXT	+VE	4	DI-EXT	+VB	4	DO	+VH	6	
8	Palopelti	G313	PP16	4	DI-EXT	+VF	5	DI-EXT	+VC	5	DO	+VD	7	
9	Palopelti	G314	PP17	5	DI-EXT	+VD	6	DI-EXT	+VD	4	DO	+VD	7	
10	Palopelti	G315	PP18	5	DI-EXT	+VH	3	DI-EXT	+VF	7	DO	+VH	11	
11	Palopelti	G316	PP19	5	DI-EXT	+VA	4	DI-EXT	+VD	9	DO	+VC	12	
12	Palopelti	G317	PP20	5	DI-EXT	+VD	5	DI-EXT	+VF	15	DO	+VF	13	
13	Palopelti	G318	PP21	6	DI-EXT	+VE	2	DI-EXT	+VE	7	DO	+VA	3	
14	Palopelti	G319	PP22	6	DI-EXT	+VD	3	DI-EXT	+VB	8	DO	+VF	4	
15	Palopelti	G320	PP23	7	DI-EXT	+VG	4	DI-EXT	+VC	9	DO	+VD	5	
16	Palopelti	G321	PP24	7	DI-EXT	+VH	5	DI-EXT	+VD	10	DO	+VC	6	
17														

KUVA 10. Esimerkki täytetystä pisteluettelosta (Pekkarila, 2019)

Valitsin kaksoisarvojen tarkistamiseen taulukkomuotoisen tiedon, koska taulukon sisällä on helpompi suorittaa taulukon alkioiden vertailu edellä kerrotulla tavalla. Lisäksi täytyisi vielä toteuttaa jokin menetelmä, miten kaksoiskappaleet tunnistetaan ja merkitään. Tässä otin käyttöön vielä toisen kolmiulotteisen taulukon virheet(k,i,j). Toteutin koodin siten, että taulukoita arvot() ja virheet() selataan rinnakkain ja aina, kun vertailtavat luvut ovat yhtä suuret, niin tämä taulukon virheet() alkio saa arvon 1. Näin voidaan jälkikäteen selvittää, missä soluissa samoja arvoja esiintyy. Lopuksi on suhteellisen helppoa käsitellä Pisteluettelo-taulukkoa siten, että samoja arvoja sisältävien solujen taustaväri muuttuu punaiseksi (kuva 11). Edellä olevilla testiarvoilla tulos on seuraavan kuvan mukainen.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	Lajittele ja tarkista			Tee kytkentäkuva										
1														
2	Laite	Positio	Moduuli (osoite)	Pisteet										
3	Palopelti	G308	PP17	2	DI-EXT	+VE	3	DI-EXT	+VH	11	DO	+VE	7	
4	Palopelti	G309	PP16	2	DI-EXT	+VB	8	DI-EXT	+VG	4	DO	+VG	8	
5	Palopelti	G310	PP15	3	DI-EXT	+VE	1	DI-EXT	+VB	13	DO	+VF	11	
6	Palopelti	G311	PP14	3	DI-EXT	+VD	3	DI-EXT	+VA	3	DO	+VD	5	
7	Palopelti	G312	PP15	4	DI-EXT	+VE	4	DI-EXT	+VB	4	DO	+VH	6	
8	Palopelti	G313	PP16	4	DI-EXT	+VF	5	DI-EXT	+VC	5	DO	+VD	7	
9	Palopelti	G315	PP18	5	DI-EXT	+VH	3	DI-EXT	+VF	7	DO	+VH	11	
10	Palopelti	G316	PP19	5	DI-EXT	+VA	4	DI-EXT	+VD	9	DO	+VC	12	
11	Palopelti	G317	PP20	5	DI-EXT	+VD	5	DI-EXT	+VF	15	DO	+VF	13	
12	Palopelti	G314	PP17	5	DI-EXT	+VD	6	DI-EXT	+VD	4	DO	+VD	7	
13	Palopelti	G318	PP21	6	DI-EXT	+VE	2	DI-EXT	+VE	7	DO	+VA	3	
14	Palopelti	G319	PP22	6	DI-EXT	+VD	3	DI-EXT	+VB	8	DO	+VF	4	
15	Palopelti	G320	PP23	7	DI-EXT	+VG	4	DI-EXT	+VC	9	DO	+VD	5	
16	Palopelti	G321	PP24	7	DI-EXT	+VH	5	DI-EXT	+VD	10	DO	+VC	6	

KUVA 11. Esimerkki Pisteluettelo-taulukosta tarkastuksen jälkeen (Pekkarila, 2019)

Jos ohjelma hälyttää kaksoisarvoista (solun taustaväri on punainen), niin virheelliset arvot täytyy korjata ja tämän jälkeen suorittaa tarkistus uudestaan, painamalla Lajittele ja tarkista -komentopainiketta.

Kun Pisteluettelo-taulukko on lajiteltu ja tarkastettu (ja mahdolliset virheet korjattu), voidaan muodostaa KytKentäkuva-taulukko. Tämä taulukko sisältää kunkin moduulin yksityiskohtaiset tiedot, jotka ovat omilla sivuillaan. Taulukko on suunniteltu siten, että tulostettaessa käytetään vaakasuuntaista tulostusasetusta. Lisäksi rakenne on sellainen, että jokaisen sivun pituus on 35 riviä. Näin jokainen moduuli mahtuu omalle sivulleen.

KytKentäkuvien tuottaminen Pisteluettelo-taulukosta on siinä mielessä yksinkertaista, että kyseessä on tiedonsiirto kahden taulukon välillä. Ongelmana on varmistaa, että tiedot siirtyvät Pisteluettelo-taulukon oikeista soluista KytKentäkuva-taulukon oikeisiin soluihin.

Ohjelmassa tarvitaan edellä mainittuja taulukoita moduli() ja rivit(). Lisäksi tarvitaan taulukko positio(). Tämän taulukon tietotyyppinä on String (merkkijono), koska se arvot saadaan yhdistelmänä Pisteluettelo-taulukon sarakkeissa B-D olevista arvoista, jotka ovat merkkijonoja. Seuraavassa ovat ohjelmarivit, joilla muodostetaan positio()-taulukon arvot.

```
Dim positio(100) As String
Worksheets("Pisteluettelo").Select
r = Cells(Rows.Count, 1).End(xlUp).Row
For i = 3 To r
    positio(i - 2) = Cells(i, 2).Value & Cells(i, 3).Value & Cells(i,
4).Value
Next i
```

Olen ottanut esille edellä olevan koodin esimerkkinä siitä, millä tavalla voidaan hyödyntää Excel VBA:n toistorakennetta For-Next sekä Cells-ominaisuutta. Cells-ominaisuuden avulla voidaan helposti numeerisesti valita oikeat solut tai solualueet.

KytKentäkuvat- taulukon muodostamisessa käytetään aputaulukkoa Pohja1, joka nimensä mukaisesti on kunkin kytKentäkuvan pohja. Tämä aputaulukko kopioidaan KytKentäkuvat- taulukkoon niin monta kertaa kuin eri moduuleita on järjestelmässä. Jokaisen kopioinnin jälkeen tehdään sivunvaihto. Tällä tavalla jokainen sivu käsittää 35 riviä. Näin Pisteluettelo- taulukosta tietojen siirto KytKentäkuvat- taulukkoon voidaan skaalata, jotta tiedot siirtyvät oikeille sivuille ja oikeisiin soluihin. Kun KytKentäkuvat- taulukossa siirrytään toiselle sivulle, niin riviosoitteiden arvoja kasvatetaan 35 rivillä. Ohjelmakoodin rivit, joilla tiedot siirretään, ovat liitteessä 3.

Toteutettua työkalua on pyritty testaamaan erilaisissa käyttötilanteissa ja mahdollisimman laajalla testiaineistolla. Testattaessa ohjelmakoodiin jouduttiin tekemään joitakin muutoksia. Samalla muokattiin myös taulukoiden ulkoasua ja muutenkin käyttöliittymää.

Yhteenveto työkalun toimintaperiaatteesta on kaaviokuvana liitteessä 4.

4.4.3 Saadut edut työkalusta

Työkalu vastaa sille asetetut tavoitteet ja vaatimukset. Sen avulla voidaan tuottaa tarvittavat suunnitteludokumentit, joiden perusteella järjestelmä voidaan toteuttaa. Työkalu standardisoi suunnittelun etenemisen ja nopeuttaa suunnittelua sekä vähentää suunnittelun aikaisia virheitä. Pelkästään että virheet vähenevät, myös se etu uudesta työkalusta on, jos tarvitsee tarkistaa jonkin palopellin kytKentäpiste, niin se on nopeampaa, kun on pisteluettelo kaikista palopelleistä. Pisteluettelon avulla voidaan nopeasti tarkistaa jonkin tietyn palopellin kytKentäpiste, ilman että tarvitsee alkaa selaamaan montaa sivua kytKentäkuvia.

5 POHDINTA

Opinnäytetyöni aiheen sain siis Bithouse Oy:stä. Aihe alkoi muotoutua työskentelyni kautta. Alun perin ajateltu aihe oli toteuttaa järjestelmän käyttöohje. Yrityksen kannalta oli kuitenkin hyödyllisempää ryhtyä laatimaan uutta suunnittelutyökalua, joka tehostaisi järjestelmän suunnitteluprosessia. Näin työkalun suunnittelemisesta ja toteuttamisesta tuli opinnäytetyöni aihe. Opinnäytetyötä tehdessäni sain suhteellisen vapaasti toteuttaa ideoitani. Tämäkin innosti opinnäytetyön tekemistä.

Aihe vaikutti mielenkiintoiselta ja samalla myös haastavalta, koska työn toteuttaminen edellytti mm. ohjelmointia, jota minulla on ollut suhteellisen vähän opintojeni aikana. Ohjelmointiin sain opastusta alkuvaiheessa ja jonkin verran työn aikana kollegoiltani. Excel VBA opastusta ja neuvoja sain myös DI Vesa Pekkarilalta. Lisäksi täytyi myös perehtyä palopelteihin liittyvään lainsäädäntöön. Käytännön projekteja toteuttaessani on tärkeää tuntea viranomaisten antamat määräykset ja ohjeet.

Yrityksessä päätettiin, että suunnitteludokumentit perustuvat edelleen Excel- taulukkolaskentaan, kuitenkin siten että taulukoiden käyttöä automatisoidaan Excel VBA -ohjelmointia hyödyntäen. Aluksi suurimmat ongelmani olivat ohjelmoinnin opettelussa ja yleensä siinä, miten VBA-ohjelmointia voitaisiin käyttää tässä tapauksessa. Internetistä löytyi hyvin paljon ohjelmaesimerkkejä, mutta niistä ei luonnollisestikaan ollut suoranaisesti apua. Runsaasti aikaa kuluihin ohjelmaesimerkkeihin tutustumiseen. Näiden esimerkkien avulla ja käyttämällä Excel-makrojen nauhoittamista, alkoivat ongelmat vähitellen ratketa.

Työn aikana opin monenlaista ongelmanratkaisua. Tärkeätä oli myös oppia jäsentämään ongelmat pienempiin kokonaisuuksiin, ja hakea näihin ratkaisua. Makrojen nauhoittaminen oli tästä hyvä esimerkki. Perehdyin myös lainsäädäntöön, mistä minulle on tulevaisuudessa hyötyä.

Työn lopputulokseen olen tyytyväinen. Suunnittelutyökalusta tuli sille asetettujen tavoitteiden ja vaatimusten mukainen. Pystyn hyödyntämään työkalua tulevissa

töissäni. Tosin tämä on suunnittelutyökalun ensimmäinen versio ja sen kehittäminen jatkuu.

Jouduin toteuttamaan työkalun suhteellisen nopeassa aikataulussa, mistä syystä ohjelmakoodi kaipaisi ehkä muokkaamista. Ohjelmissa voisi myös hyödyntää tehokkaammin Excel VBA:n objektien ominaisuuksia ja erilaisia ohjelmarakenteita.

LÄHTEET

ActiWeb-ohjelmisto. 2019. Bithouse Oy. Luettu 31.3.2019.
<https://bithouse.fi/actiweb-ohjelmisto/>

Ashrae. 2012. BACnet, a Data Communication Protocol for Building Automation and Control Networks. Atlanta USA: American Society of Heating, Refrigerating and Air-Conditioning Engineers.

Bithouse Actiweb käyttöönotto ja konfigurointi. 9.4.2018. Bithouse Oy. Luettu 1.4.2019. Vaatii käyttöoikeuden.

Bithouse Oy. 2019. Bithouse. Luettu 10.2.2019. <http://bithouse.fi/>

Erilaisia silmukkarakenteita. 3.2.2005. Jyväskylän yliopiston IT-tiedekunta ja avoin yliopisto. Luettu 17.3.2019.
<http://appro.mit.jyu.fi/doc/tiedonhallinta/vba/#for>. Viitattu sivulla 25.

ETS NORD Oy. 2019a. FDMS Palopelti. Luettu 16.3.2019.
https://www.etsnord.fi/wp-content/uploads/2017/08/05_FDMS_NORDfire_fi_1217.pdf

ETS NORD Oy. 2019b. ETS Control palopeltien ohjaus- ja valvontajärjestelmä. Luettu 15.3.2019. <https://www.etsnord.fi/wp-content/uploads/2018/06/ETS-Control-esite-062018.pdf>

EXCEL VBA-OHJELMOINTI. 27.9.2013. Aki Taanila. Luettu 27.3.2019.
<http://myy.haaga-helia.fi/~taaak/vba/vba.pdf>

Finlex. 21.12.2012. Maankäyttö- ja rakennuslaki. Luettu 27.2.2019.
<https://www.finlex.fi/fi/laki/ajantasa/1999/19990132#L17P117b>

Finlex. 29.4.2011. Pelastuslaki. Luettu 3.3.2019.
<https://www.finlex.fi/fi/laki/ajantasa/2011/20110379#L3P12>

Ilmanvaihtolaitteistojen paloturvallisuusopas. 1.2012. Suomen LVI-liitto. Luettu 3.4.2019. <https://docplayer.fi/917659-Julkaisu-1-2012-ilmanvaihtolaitteistojen-paloturvallisuusopas.html>

Monipuolinen Magio 816-moduuli. 2018. Bithouse Oy. Luettu 18.3.2019.
<http://bithouse.fi/magio/>

SFS-EN 15650. 2010. RAKENNUSTEN ILMANVAIHTO. PALONRAJOITTIMET. Helsinki: Suomen Standardoimisliitto SFS. Luettu 8.4.2019. Vaatii käyttöoikeuden.
<https://sales.sfs.fi/fi/index/tuotteet/SFS/CEN/ID2/1/296123.html.stx>

Sten, T. Palotarkastusinsinööri. 2019. RAKENTEELLINEN PALOTURVALLISUUS. Luento. 13.3.2019. Tampereen ammattikorkeakoulu. Tampere.

Sähkötieto ry. 2014. ST 666.09 PALOPELTIEN OHJAUS. Julkaistu 15.9.2014. Luettu 16.2.2019. 66609.pdf (suojattu)

Talotekniikkainfo. 2019. 9.3 Palopeltien käyttö. Talotekniikkateollisuus. Luettu 1.3.2019. <https://www.talotekniikkainfo.fi/ilmanvaihtolaitosten-paloturvallisuus-opas/9-3>

Ympäristöministeriön asetus rakennusten paloturvallisuudesta. 28.11.2017. Ympäristöministeriö. Luettu 1.4.2019.

LIITTEET

Liite 1. Moduulien lukumäärän määrittäminen

```
n = 1
Worksheets("Pisteluetelo").Select
Dim moduli(100) As Integer
Dim rivit(10) As Integer
r = Cells(Rows.Count, 1).End(xlUp).Row

arvo1 = Worksheets("Pisteluetelo").Cells(3, 5)
moduli(1) = arvo1

For i = 4 To r
    arvo2 = Worksheets("Pisteluetelo").Cells(i, 5)
    If arvo2 = arvo1 Then
        GoTo seur
    Else
        n = n + 1
        arvo1 = arvo2
        moduli(n) = arvo1
        GoTo seur
    End If
seur: Next i
```

Liite 2. Kaksoisarvojen hakeminen

```
Dim arvot(100, 4, 3)
Dim virheet(100, 4, 3)
rivi = 3
For k = 1 To sivujen_maara
    m = rivit(k)
    For i = 1 To m
        arvot(k, i, 1) = Worksheets("Pisteluetelo").Cells(rivi, 8)
        arvot(k, i, 2) = Worksheets("Pisteluetelo").Cells(rivi, 11)
        arvot(k, i, 3) = Worksheets("Pisteluetelo").Cells(rivi, 14)
        rivi = rivi + 1
    Next i
Next k
```

Liite 3. Tietojen siirtäminen KytKentäKuvat-taulukkoon

```

' kopioidaan tiedot Pisteluettelo-taulukosta KytKentäKuvat-taulukkoon
k = 0
m = 3
n = 0
'rivi = 1
p = 1

For sivu = 1 To sivujen_maara

    rivi = rivit(sivu)

    For i = 7 + k To 8 + k
        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pisteluet-
ettelo").Cells(m, i - k)
    Next i
    t = i - 2
    For i = 9 + k To 10 + k
        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pisteluet-
ettelo").Cells(m, i + 1 - k)
    Next i
    For i = 11 + k To 12 + k
        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pisteluet-
ettelo").Cells(m, i + 2 - k)
    Next i
    Worksheets("KytKentäKuvat").Cells(t, 9) = positio(p)
    p = p + 1
    m = m + 1

    If rivi = 1 Then          'jos moduulin rivejä on vain yksi, niin
hypätään seuraavalle sivulle
        GoTo seur_sivu
    End If

    For i = 14 + k To 15 + k
        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pisteluet-
ettelo").Cells(m, i - 7 - k)
    Next i
    t = i - 2
    For i = 16 + k To 17 + k
        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pisteluet-
ettelo").Cells(m, i - 6 - k)
    Next i
    For i = 18 + k To 19 + k
        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pisteluet-
ettelo").Cells(m, i - 5 - k)
    Next i
    Worksheets("KytKentäKuvat").Cells(t, 9) = positio(p)
    p = p + 1
    m = m + 1

    If rivi = 2 Then          'jos moduulin rivejä on vain kaksi, niin
hypätään seuraavalle sivulle
        GoTo seur_sivu
    End If

    For i = 21 + k To 22 + k
        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pisteluet-
ettelo").Cells(m, i - 14 - k)
    Next i
    t = i - 2
    For i = 23 + k To 24 + k

```

```

        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pistel-
ettelo").Cells(m, i - 13 - k)
    Next i
    For i = 25 + k To 26 + k
        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pistel-
ettelo").Cells(m, i - 12 - k)
    Next i
    Worksheets("KytKentäKuvat").Cells(t, 9) = positio(p)
    p = p + 1
    m = m + 1

    If rivi = 3 Then          'jos moduulin rivejä on vain kolme, niin
hypätään seuraavalle sivulle
        GoTo seur_sivu
    End If

    For i = 28 + k To 29 + k
        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pistel-
ettelo").Cells(m, i - 21 - k)
    Next i
    t = i - 2
    For i = 30 + k To 31 + k
        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pistel-
ettelo").Cells(m, i - 20 - k)
    Next i
    For i = 32 + k To 33 + k
        Worksheets("KytKentäKuvat").Cells(i, 2) = Worksheets("Pistel-
ettelo").Cells(m, i - 19 - k)
    Next i
    Worksheets("KytKentäKuvat").Cells(t, 9) = positio(p)
    p = p + 1
    m = m + 1

    'Worksheets("KytKentäKuvat").Cells(3 + k, 9) = Worksheets("Pistel-
uettelo").Cells(3 + m, 5)
seur_sivu:

    k = k + 35          'siirrytään taulukossa 35 riviä eteen päin

Next sivu

```

Liite 4. Kaaviokuva työkalun toimintaperiaatteesta

Tee pisteluettelo		
Laitetyyppi	Positto	Tunnus
Palopelti	G308	PP17
Palopelti	G309	PP18
Palopelti	G310	PP19
Palopelti	G311	PP20
Palopelti	G312	PP21
Palopelti	G313	PP22
Palopelti	G314	PP23
Palopelti	G315	PP24
Palopelti	G316	PP25
Palopelti	G317	PP26
Palopelti	G318	PP27
Palopelti	G319	PP28
Palopelti	G320	PP29
Palopelti	G321	PP30
Palopelti	G322	PP31

Laitetiedot-taulukko

Kopioidaan laitetiedot

Lajittele ja tarkista		Tee kytkentäkuva						
Laitte	Positto	Moduuli (osoite)		Pisteet				
Palopelti	G308	2	DI-EXT +VE	3	DI-EXT +VH	11	DO +VE	7
Palopelti	G309	2	DI-EXT +VB	8	DI-EXT +VG	4	DO +VG	9
Palopelti	G310	3	DI-EXT +VE	1	DI-EXT +VB	13	DO +VF	11
Palopelti	G311	3	DI-EXT +VD	3	DI-EXT +VA	4	DO +VD	5
Palopelti	G312	4	DI-EXT +VE	4	DI-EXT +VB	10	DO +VH	6
Palopelti	G313	4	DI-EXT +VF	5	DI-EXT +VC	11	DO +VD	7
Palopelti	G315	5	DI-EXT +VH	3	DI-EXT +VF	7	DO +VH	11
Palopelti	G316	5	DI-EXT +VA	4	DI-EXT +VD	9	DO +VC	12
Palopelti	G317	5	DI-EXT +VD	5	DI-EXT +VF	15	DO +VF	13
Palopelti	G314	5	DI-EXT +VD	6	DI-EXT +VD	1	DO +VD	2
Palopelti	G318	6	DI-EXT +VE	2	DI-EXT +VE	7	DO +VA	3
Palopelti	G319	6	DI-EXT +VD	9	DI-EXT +VB	8	DO +VF	4
Palopelti	G320	7	DI-EXT +VG	4	DI-EXT +VC	2	DO +VD	11
Palopelti	G321	7	DI-EXT +VH	5	DI-EXT +VD	10	DO +VC	6
Palopelti	G322	8	DI-EXT +VB	6	DI-EXT +VE	11	DO +VA	7

Pisteluettelo-taulukko

Tarkistetaan ja kootaan tiedot

KOHDE:	SISÄLTÖ:	REV.	PVM.	Suunnittelija
BIT HOUSE	Keskus x Moduuli os:			
	Moduuli os: 5			

Liitin	Tunnus	Nomak 4x2x0,5+0,5	Palopelti	Kytk.	Tent.	Huomio
+VH		1or	S1			
3		1va	S2			
+VF		2or	S4			
7		2va	S6			
+VH		3or	1/PU			
11		3va	2/MU			
Kaapeli: G315PP24						
+VA		1or	S1			
4		1va	S2			
+VD		2or	S4			
9		2va	S6			
+VC		3or	1/PU			
12		3va	2/MU			
Kaapeli: G316PP25						
+VD		1or	S1			
5		1va	S2			
+VF		2or	S4			
15		2va	S6			
+VF		3or	1/PU			
13		3va	2/MU			
Kaapeli: G317PP26						
+VD		1or	S1			
6		1va	S2			
+VD		2or	S4			
1		2va	S6			
+VD		3or	1/PU			
2		3va	2/MU			
Kaapeli: G318PP23						

Kytkentäkuvat-taulukko