Jussi Saari

Ohjaustyökalujen suunnittelu ja kehitys

Opinnäytettyö
Kevät 2015
SeAMK Tekniikan yksikkö
Bio- ja elintarviketekniikan koulutusohjelma
Opinnäytetyön tiivistelmä

Koulutusyksikkö: Tekniikka
Koulutusohjelma: Bio- ja elintarviketekniikka
Suuntautumisvaihtoehto: Lihateknologia
Tekijä: Jussi Saari
Työn nimi: Ohjaustyökalujen suunnittelu ja kehitys
Ohjaaja: Jarmo Alarinta
Vuosi: 2015 Sivumäärä: 32 Liitteiden lukumäärä: 0


Avainsanat: tuotannonohjaus, Excel, Visual Basic for Applications
This thesis presents in the form of a report on the process of creating resource planning tools for the company that ordered the work. The objective of the work was not only to fulfil the requirements that the company had for this work but also to create a basis for the other resource planners at the company for creating similar tools. This thesis also covers the use of Microsoft Excel functions and VBA-programming.

The main work that was carried out for the company was a resource planning tool that included daily production amounts for all the company’s products for 4 weeks ahead. These production amounts can be used to improve the company’s resource handling and machine capacity. The work was completed with the help of Microsoft Excel.

Keywords: resource planning, Excel, Visual Basic for Applications
SISÄLTÖ

Opinnäytetyön tiivistelmä.................................................................2
Thesis abstract.............................................................................3
SISÄLTÖ ...........................................................................................4
Käytetyt termit ja lyhenteet .........................................................6
1 JOHDANTO .................................................................................7
2 TAUSTATIETOA ..........................................................................8
  2.1 Tuotannonsuunnittelu.............................................................8
  2.2 Tietovarastot ........................................................................10
  2.3 Microsoft Excel –funktiot .......................................................11
    2.3.1 PHAKU ja VHAKU............................................................11
    2.3.2 SUMMA.JOS....................................................................13
    2.3.3 INDEKSI .........................................................................14
    2.3.4 VASTINE .........................................................................15
    2.3.5 Matriisikaavat .................................................................16
  2.4 Microsoft Excel - VBA .........................................................17
    2.4.1 Esimerkkiohjelmointi................................................................17
3 TYÖN TOTEUTUS .....................................................................22
  3.1 Lähtökohdat .........................................................................22
  3.2 Toimintaan perehtyminen......................................................22
  3.3 Työn suunnittelu ....................................................................23
  3.4 Suunnittelutyökalujen valmistus ...........................................24
    3.4.1 Tuotetiedot .......................................................................24
    3.4.2 Myynti- ja ennustetietojen haku tietovarastoraportteista ......25
    3.4.3 Varastosaldon laskeminen saldoraportista ..........................27
    3.4.4 Tuotantosuunnitelma .......................................................28
    3.4.5 Tuotantolistat .................................................................29
4 TYÖN TARKASTELU ..................................................................30
LÄHTEET .........................................................................................31

Kuvio- ja taulukkoluettelo
Kuvio 1. Tilaus-toimitusprosessi ................................................................. 9
Kuvio 2. Tuotannonsuunnitelin pääohdat .................................................. 10
Kuvio 3. Tietovarastoraportin luominen eri vaiheet .................................. 11
Kuvio 4. PHAKU ...................................................................................... 12
Kuvio 5. SUMMA.JOS ................................................................................ 14
Kuvio 6. INDEKSI .................................................................................... 15
Kuvio 7. VASTINE .................................................................................... 16
Kuvio 8. PHAKU-funktion korvaus INDEKSI- ja VASTINE-funktioiden avulla .. 16
Kuvio 9. Esimerkki tavallisen kaavan korvaamista matriisikaavalla ................... 17
Kuvio 10. Tuotantosuunnitelma edellisenä iltana ennen makron käyttöä ........ 18
Kuvio 11. Tuotantosuunnitelma seuraavana aamuna makron käyttämisen jälkeen ........................................................................................................... 18
Kuvio 12. Esimerkki valmiasta ohjelmointikoodista .................................... 21
Kuvio 13. Asiakassuhteet .......................................................................... 23
Kuvio 14. Hierarkia .................................................................................... 24
Kuvio 15. Tuotetiedot ............................................................................... 25
Kuvio 16. Tietovarastoraportin esimerkkinäkymä ....................................... 26
Kuvio 17. Tietovarastoraportin esimerkkimuunto ....................................... 27
Kuvio 18. Saldolaskenta ........................................................................... 28
Kuvio 19. Vähimmäisvalmistuksen laskenta ............................................... 28
Kuvio 20. Tuotantosuunnitelma ................................................................ 29
Kuvio 21. Tuotantolista, jossa ennuste korjataan tarvittaessa lisätilauksella .... 29
### Käytetyt termit ja lyhenteet

<table>
<thead>
<tr>
<th>Termi</th>
<th>Käännös</th>
</tr>
</thead>
<tbody>
<tr>
<td>VBA</td>
<td>Visual Basic for Applications. Microsoftin kehitämä ohjelmointikieli.</td>
</tr>
<tr>
<td>Lean</td>
<td>Ajattelufilosofia, joka pyrkii arvoa tuottamattomien toimintojen minimointiin.</td>
</tr>
</tbody>
</table>
1 JOHDANTO

Opinnäytetyöni sai alkunsa yrityksen tuotannonohjauksen kehitystarpeesta, jossa haluttiin laajentaa suunnittelun aikaikkunaa muutamasta päivästä useisiin viikoihin. Erityisesti uusilla työkaluilla haluttiin kehittää yrityksen materiaali-, raaka-aine- ja resurssihallintaa kohti Lean-ajattelumallin johtamisfilosofiaa.


Tämän opinnäytetyön tarkoitus on antaa lukijalle selkeä yleiskuva tuotannonohjauksen työkalujen valmistamisesta sekä toimia yritykselle ohjeena työkalujen tarkempaan tarkasteluun. Työ koostuu lyhyestä tuotannonohjauksen esittelystä, työssä käytettyjen Excel-funktioiden sekä VBA-ohjelmoinnin esittelystä, ohjemaisesta raportista työkalujen valmistamisesta ja tulososiosta.
2 TAUSTATIETOA

2.1 Tuotannonsuunnittelu

Hyvin toimiva tuotanto on monien tekijöiden summa, ja yksi tärkeimmistä tekijöistä on tuotannonsuunnittelu. Tuotannonsuunnittelu lähteekin liikkeelle jo tuotteen kehitysvaiheessa, jolloin on osattava ottaa kantaa kehitettävän tuotteen ominaisuuksiin, kuten esimerkiksi säilyvyyteen ja valmistusprosessiin.

Tilaus-toimitusprosessi voidaan usein yksinkertaistaa tavarantoimittajan, yrityksen ja asiakkaan väliseksi vuorovaikutukseksi (Sakki 2014). Tässä vuorovaikutuksessa siirtyy tavaran ja rahan lisäksi myös tietoa (Sakki 2014). Tilaus-toimitusprosesssin käynnistymisen impulssi voi olla joko tilaus tai ennuste riippuen yrityksen tuotteiden ominaisuuksista ja sovistusta toimitusrytmistä. Tämän vuoksi rajaveto ennusteeseen ja tilaukseen perustuva toimintoon on hankala tehdä, sillä ei ole olemassa yhtä oikeaa toimintamallia.
Kuvio 1. Tilaus-toimitusprosessi

Kysynnän ennustaminen tehdään jokaiselle tuotteelle erikseen päiväkohtaisesti huomioiden mahdolliset kysyntäpiikit. Jos halutaan tarkentaa ennustetta vielä tästä, on tuotteen päiväkohtainen kysyntä purettava vielä asiakas- tai ketjutasolle. Ennustaminen voidaan tehdä esimerkiksi edellisten viikkojen keskiarvo-laskuna (Sakki

![Diagram](image)

Kuvio 2. Tuotannonsuunnittelun pääkohdat

### 2.2 Tietovarastot

Toiminnanohjausjärjestelmää käytettäessä syntyy jatkuvasti lisää informaatiota, jonka seuraaminen antaa paitsi työntekijälle myös yritykselle arvokasta lisätietoa. Tämä tieto muutetaan ETL-vaiheessa (Extract – Transform – Load) tietovarastokannan vaatimaan muotoon (Hovi, Hervonen & Koistinen 2009, 14). ETL-vaihe voidaan
toteuttaa paitsi halutulla taajuudella myös reaaliaikaisesti (mts. 29). Tietovarastoraporttien kautta tietovarastokantaan siirretty tieto saadaan työntekijän ulottuville joko valmiissa tai taulukkolaskentaohjelmalla jatkojalostettavassa muodossa (mts. 6-7).

Kuvio 3. Tietovarastoraportin luominen eri vaiheet

2.3 Microsoft Excel –funktiot


2.3.1 PHAKU ja VHAKU

PHAKU-funktion avulla voidaan etsiä määrittelyn matriisin ensimmäisestä sarakkeesta hakuarvoa vastaavia arvoja ja palauttaa samalta riviltä halutun sarakkeen
arvon. Funktio muodostetaan neljästä argumentista, jotka ovat hakuarvo, matriisi, sarakkeen indeksinumero ja totuusarvo. Kaava kirjoitetaan tällöin muotoon PHAKU (hakuarvo ; matriisi ; sarakkeen indeksinumero ; totuusarvo). Matriisin ensimmäinen sarakke toimii aina hakuarvon etsintäsarakkeena ja määrittää rivin, jolta palautettava arvo löytyy. Sarakkeen indeksinumero puolestaan kertoo, mistä sarakkeesta palautettava arvo löytyy. Totuusarvon avulla määritetään halutaanko etsiä hakuarvon tarkka vai lähin vastaava. (Microsoft, PHAKU-funktio.)

Kuviossa 4 on esitetty esimerkki PHAKU-funktion käytöstä. Tässä esimerkissä halutaan soluun G1 palauttaa arvo, joka löytyy samalta riviltä kuin hakuarvo B-sarakkeesta ja matriisin kolmannesta sarakkeesta. Sana "tiistai" löytyy B-sarakkeesta riviltä 2, ja matriisin kolmas sarakke on D, jolloin soluun G1 palautuu arvo, joka löytyy solusta D2 eli viikon toinen päivä.

![Diagram](Image)

Kuvio 4. PHAKU

PHAKU-funktio on erityisen hyödyllinen, kun hakumatriisin ensimmäinen sarakke sisältää jokaista arvoa vain yhden. Jos ensimmäinen sarakke sisältää hakuarvoa vastaavia arvoja enemmän kuin yhden, palauttaa funktio pienemmältä riviltä lötyvän arvon.
VHAKU-funktio on hyvin samankaltainen funktio kuin PHAKU. Erotuksena näillä kahdella on se, että VHAKU etsii hakuarvoa matriisin ylimmältä riviltä, jolloin saadaan selville, mistä sarakkeesta palautettava arvo löytyy. Rivinumero määritetään rivin indeksinumeron avulla. (Microsoft, VHAKU-funktio.)

Hyvä esimerkki PHAKU- ja VHAKU-funktioiden käytöstä suunnittelutyökaluissa on tiedon tai ominaisuuden tuotekoodin viereiseen soluun laskentaa tai informaatiota varten.

2.3.2 SUMMA.JOS

SUMMA.JOS-funktion avulla saadaan laskettua yhteen sarakkeen arvoja määritetyjen ehtojen avulla. Syntaksin argumentit ovat hakualue, hakuehto ja summa-alue. Kaava kirjoitetaan tällöin muotoon SUMMA.JOS(hakualue ; hakuehto ; summa-alue. Hakuehtona ei tarvitse välttämättä olla yksi sana tai numero, vaan ehtona voidaan käyttää myös suurempi kuin- ja pienempi kuin-lauseita. (Microsoft, SUMMA.JOS-funktio.)

SUMMA.JOS-funktio on erittäin hyödyllinen, jos halutaan esimerkiksi summata yrityksen kaikkien tuotteiden eri materiaalien päivä- tai viikkokohtaiset kulutukset.

### 2.3.3 INDEKSI

INDEKSI-funktion avulla voidaan palauttaa halutusta hakumatriisista rivi- ja sarakenumeron perusteella solun arvo. Funktioin argumentit ovat hakumatriisi, rivinumero ja sarakenumero. Kaava kirjoitetaan muotoon INDEKSI(hakumatriisi ; rivinumero ; sarakenumero). (Microsoft, INDEKSI-funktio.)

2.3.4 VASTINE

VASTINE-funktion avulla voidaan etsiä määritellystä hakumatriisista hakuarvoa vastaavan arvon suhteellinen sijainti matriisissa. Funktion argumentit ovat hakuarvo, hakumatriisi ja vastinelaji. Kaava kirjoitetaan muotoon VASTINE(hakuarvo ; hakumatriisi ; vastinelaji). Vastinelajin avulla määritellään halutaanko etsiä hakuarvon tarkkaa vastinetta (0 tai puuttuu), lähintä pienempää tai yhtä suurta vastinetta (-1) vai lähintä suurempaa tai yhtä suurta vastinetta (1). (Microsoft, VASTINE-funktio.)

Kuvion 7 solun E5 kaavassa etsitään solun A5 arvon suhteellista sijaintia hakumatriisista 1:1. Arvo F löytyy hakumatriisin kuudennesta sarakkeesta, joten Funktio palauttaa arvon 6. Vastinelaji ei tässä tapauksessa voi olla mikään muu kuin 0, sillä vastinelajit 1 ja -1 toimivat halutulla tavalla vain numeroita käytettäessä.

Kuvio 6. INDEKSI
VASTINE-funktio on erinomainen käyttötäväksi toisten funktioiden kanssa. Esimerkiksi korvaamalla INDEKSI-funktion rivinumero argumentin VASTINE-funktioilla, saadaan aikaan hyvin pitkälti PHAKU-funktion kaltainen kaava. Tällä yhdistelmällä pystyy hakemaan arvoja jopa PHAKU-funktioniota joustavammin, sillä lukuja voidaan hakea myös vasemmalle päin edetien, kuten kuvion 8 esimerkissä on esitetty.

Kuvio 8. PHAKU-funktion korvaus INDEKSI- ja VASTINE-funktioiden avulla

2.3.5 Matriisikaavat

Matriisikaavojen avulla voidaan suorittaa hankalia laskutoimituksia, joiden ratkaisu tavanomaisia kaavoja käyttämällä vaatisi useita eri väli vaiheita, sillä matriisikaava osaa suorittaa useita eri laskutoimituksia samaan aikaan. Matriisikaavan kirjoittami-
nen jälkeen on muistettava aina painaa CTRL+VAIHTO+ENTER, sillä muutoin kirjoitettu kaava jää tavalliseksi kaavaksi eikä toimi halutulla tavalla. (Microsoft, Matriisikaavoihin liittyviä ohjeita ja esimerkkejä.)

Kuvion 9 sarakkeessa D näkyvät kaavat, joilla sarakkeiden A-C osoittamat laskutoimitukset voidaan suorittaa tavanomaisia kaavoja käyttämällä. Sarakkeessa E on puolestaan esitetty samat laskutoimitukset matriisikaavojen käyttäen.

Kuvio 9. Esimerkki tavallisen kaavan korvaamista matriisikaavalla

### 2.4 Microsoft Excel - VBA

VBA-ohjelmointia voidaan suorittaa joko nauhoittamalla toimenpidesarja tai kirjoittamalla itse ohjelmointikoodia. Nauhoittamalla kirjoitettujen ohjelmointikoodien rajotteet tulevat vastaan suhteellisen nopeasti, kun taas itse kirjoitettaessa nuo rajat ovat huomattavasti laajemmat. Ohjelmointiratkaisu on syytä muistaa, että ohjelmointi on itseasiassa enemmän ongelmanratkaisua kuin ohjelmoinnin opettelua. (Taanila 2013, 1.) Hyvä tapa opiskella ohjelmointikieltä on seurata nauhoitettuja koodinpalasia (mts. 6).

#### 2.4.1 Esimerkkiohjelmointi

Seuraavassa esimerkissä esitellään ohjelmointikoodi, jonka avulla voidaan toteuttaa haluttuja toimenpiteitä määrittelyjen ehtojen täyttymisä. Koodi on erityisen hyödyllinen korvaamaan muutoin käsityönä tehtävää virheeltä ja toistuvaa työtä.
Kuviossa 10 on esitetty kahden tuotteen tuotantosuunnitelmat, jotka halutaan siirtää seuraavan päivän aamuna vastaamaan eilen tehtyjä suunnitelmia. Kuviossa 11 on sama taulukko koodin suorittamisen jälkeen.

Kuviota.

Kuviota.

Kuviosta 10. Tuotantosuunnitelma edellisenä iltana ennen makron käyttöä

Kuviosta 11. Tuotantosuunnitelma seuraavana aamuna makron käyttämisen jälkeen
Kuviosta 12 löytyy valmis ohjelmointikoodi, joka suorittaa esimerkissä mainitut tehtävät. Alla on selitetty eri osoiden suorittamat tehtävät.

**Application.ScreenUpdating = False**

**Application.Calculation = xlCalculationManual**

- Ruudun päivityksen ja laskennan poistaminen käytöstä laskennan nopeuttamiseksi

**Windows("Mallityökalu.xlsm").Activate**

**Sheets("TUOTANTOSUUNNITELMA").Select**

**Range("C1").Select**

- Valitsee avuksi olevista työkirjoista Mallityökalu.xlsm:n TUOTANTOSUUNNITELMA-välilehden C1-solun

**Do**

  **If Range("C" & ActiveCell.Row) = "SUUNNITELMA (KG)" Then**

  **ActiveCell.Offset(0, 2).Resize(1, 2).Select**

  **Selection.Copy**

  **Range("D" & ActiveCell.Row).Select**

  **ActiveSheet.PasteSpecial Format:=3, Link:=1, DisplayAsIcon:=False, IconFileName:=False**

  **ActiveCell.Offset(0, 2).Select**

  **Selection.ClearContents**

  **Range("C" & ActiveCell.Row).Select**

  **End If**

**ActiveCell.Offset(1, 0).Select**

**Loop Until (ActiveCell.Address = "$C$500")**

- Koodin ydinosa, joka käy läpi alueen C1:C500 ja etsii tekstiä SUUNNITELMA (KG). Jos teksti löytyy, siirrytään solusta 2 solua oikealle
ja suurennetaan valintaa yhden solun verran vasemmalle. Tämän jälkeen palataan saman rivin D-sarakkeeseen, johon valittu alue liitetään. Seuraavaksi siirrytään vielä tästä 2 solua vasemmalle ja tyhjennetään valitun solun sisältö. Lopuksi palataan saman rivin C-sarakkeeseen. IF-lause päättyy täähän. Tämän jälkeen Siirrytään C-sarakkeessa seuraavalle riville ja jatketaan seuraavaa SUUNNI-TELMA (KG)-tekstin etsintään

Application.ScreenUpdating = True
Application.Calculation = xlCalculationAutomatic
- Ruudun päivityksen ja laskennan palauttaminen käyttöön
Sub Siirrä_muunnitelma()

Application.ScreenUpdating = False

Windows("Mallityökalu.xlsm"),.Activate
Sheets("TUOTANOSUUNNITELMA").Select
Range("C1").Select

Do
    If Range("C" & ActiveCell.Row) = "SUUNNITELMA (KS)" Then
        ActiveCell.Offset(0, 2).Resize(1, 2).Select
        Selection.Copy
        Range("D" & ActiveCell.Row).Select
        ActiveSheet.PasteSpecial Format:=8, Link:=1, DisplayAsIcon:=False,
        IconFileName:=False
        ActiveCell.Offset(0, 2).Select
        Selection.ClearContents
        Range("C" & ActiveCell.Row).Select
    End If
Loop Until (ActiveCell.Address = "$C$500")

Application.ScreenUpdating = True
Application.Calculation = xlCalculationAutomatic

End Sub

Kuvio 12. Esimerkki valmiista ohjelmointikoodista
3 TYÖN TOTEUTUS

3.1 Lähtökohdat

Microsoft Excel -pohjaisten suunnittelutyökalujen lähtökohtana on usein toiminnanohjausjärjestelmän hienosuunnitteluun ja muokattavuuden tarve. Esimerkiksi reaaliaikaisen varastosaldon käyttö tilausohjautuvilla lyhyen varastointiajan omaavilla tuotteilla voi olla hankalaa ja joissakin tapauksissa jopa mahdotonta. Lisäksi aloilla, joilla on laaja ja monimuotoinen tuotevalikoima, ulkoistettu toiminnanohjausjärjestelmä ei aina tarjoa sitä muokattavuutta, jota tuotannonohjausjärjestelmä tarvitsisi.

3.2 Toimintaan perehtyminen

Ennen kuin uusien suunnittelutyökalujen suunnittelu voidaan aloittaa, on suunnitellijan syytä perehtyä läpiotaisiin valmistavan osaston toimintaan. Etenkin prosesseista, käytettävistä raaka-aineista ja loppujuotteista on oltava erittäin hyvä käsitys. Lisäksi on tunnettava tavarantoimittajien ja asiakkaiden luomat vaatimukset ja rajotukset. Perehdyttämiseen saattaa kulua pitkiä aikoja, mutta ilman sitä suunnittelutyökalujen luontia on erittäin hankala edes aloittaa.
Työn suunnittelu

Valmistettaessa Microsoft Excel -pohjaisia työkaluja, on työjärjestys ja tietyhierarkia suunniteltava hyvin, sillä tällä vähennetään puutteellisen hierarkian riskiä olennaisesti. Puutteellinen hierarkia saattaa hankalimmissa tapauksissa tarkoittaa työn aloittamista jopa kokonaan uudelleen. Hyvä perusperiaate on, että tietoa ei voi olla liikaa hierarkian eri tasoilla. On kuitenkin huomioitava, että tiedon määrän lisääntyessä kokonaisuuden hahmottaminen voi vaikeutua. Siksi työkalujen valmistuksessa onkin alusta alkaen pyrittävä mahdollisimman selkeään ulosantiin.
3.4 Suunnittelutyökalujen valmistus

Tässä osiossa käydään läpi suunnittelutyökalujen valmistuksen eri vaiheet pääpiirteittäin. Kappaleen tarkoituksena on antaa yleiskuva suunnittelutyökalujen toimintaperiaatteesta, joita hyväksikäyttämällä saadaan rakennettua yrityksen tarpeisiin räätälöity tuotannonohjausjärjestelmä.

3.4.1 Tuotetiedot

Ensimmäinen työvaihe on koota tuotteen olennaiset ominaisuudet ja tiedot. Koonti on hyvä aloittaa tuotekoodien ja näitä vastaavien nimien syöttämisellä. Tämän jälkeen tuotteen ominaisuudet kerätään samalle riveille ominaisuudet selkeästi otsikoiden.
Tuotetiedot

Elintarvikealan yritykselle olennaisia ominaisuuksia ovat muun muassa:

- Säilyvyysaika
- Asiakkaan hyllyaika
- Valmistusaineet ja niiden osuus tuotteesta
- Hävikit
- Rasian sisältämä kappalemäärä
- Rasiapaino
- Myyntierä-, laatikko- ja lavakoko
- Pakkausmateriaalit ja -koodit
- Tuotantolinja

3.4.2   Myynti- ja ennustetietojen haku tietovarastoraporteista

Käytettävän tietovarastoraportin on sisällettävä vähintään suunnittelijan ohjaamien tuotteiden myynti- ja ennustetiedot päiväkohtaisesti. Lisäksi jossain tapauksissa on tuotteen päiväkohtainen myynti purettava vielä asiakastasolle, jolloin pystytään ohjaamaan tuotannon valmistamia tuotteita useille eri asiakkaille suoraan tuotantosastolta käsin ilman erillisen logistiikkakeskuksen apua.

Kuviossa 16 on esitetty havainnollistava esimerkki tietovarastoraportin tietoalueesta, jotka on muutettu kuviossa 17 näkyvien kaavojen avulle muotoon, joita on helppo käyttää jatkossa hyväksi. Erityistä huomiota on syytä kiinnittää kuvion 17 A-
sarakkeen ketjuskaavoihin, joiden lopputulema on oltava muodossa "TUOTEKOODI_PVM", jolloin tätä saraketta voidaan käyttää hakusarakkeena haettaessa tuotteen päiväkohtaista kokonaismyyntiä tai –ennustetta tuotantoennusteeeseen. Jos halutaan vielä jakaa päiväkohtainen myynti asiakastasolle, on tätä varten syytä tehdä omalle sarakkeelle ketjuskaavat, joiden lopputulema on muotoa "TUOTEKOODI_PVM_ASIAKASNUMERO". Ketjutukset tehdään omille sarakkeilleen, koska ketjutuksella "TUOTEKOODI_PVM_ASIAKASNUMERO" ei voida hakea tuotteen päiväkohtaista kokonaismyyntiä ja ketjutuksella "TUOTEKOODI_PVM" puolestaan ei saada purettua tuotteiden kokonaismyyntiä asiakastasolle.

<table>
<thead>
<tr>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>TUOTEKOODI</td>
<td>TUOTENIMI</td>
<td>PVM</td>
<td>ENNUSTE (kg)</td>
<td>TILAUS (kg)</td>
</tr>
<tr>
<td>1</td>
<td>A</td>
<td>20141018</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>A</td>
<td>20141019</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>A</td>
<td>20141020</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>A</td>
<td>20141021</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>B</td>
<td>20141018</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>B</td>
<td>20141019</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>B</td>
<td>20141020</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>C</td>
<td>20141018</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>C</td>
<td>20141019</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>D</td>
<td>20141020</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>D</td>
<td>20141021</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

Kuvio 16. Tietovarastoraportin esimerkkinäkymä
Kuvio 17. Tietovarastoraportin esimerkkimuunto

3.4.3 Varastosaldon laskeminen saldoraportista

Elintarvikkeiden varastosaldon laskemisessa on otettava huomioon, että kaikki varastossa olevat tuotteet eivät välttämättä enää täytä asiakkaalle luvatuja ehtoja. Näitä luvattuja ehtoja ovat muun muassa asiakkaalle taattu hyllyaika ja pakkauksen fyysinen kunto. Näin ollen varaston absoluuttinen arvo ei aina ole yhtä suuri kuin myytävissä oleva arvo. Lisäksi on otettava huomioon, että osa varaston sisältämistä tuotteista on jo varattu asiakkaille, jotka on myös otettava huomioon.

Alla on esitetty yksi tapa sellaisen varastosaldon laskemiseen, joka sisältää asiakkaille varattuja tuotteita. Tässä tapauksessa asiakkaille varattujen tuotteiden määrä on yhtä suuri kuin seuraavan toimituspäivän tilauskanta. Tällöin varaston absoluuttisesta arvosta on vähennettävä tämä tilauskanta ja liian lyhyen hyllyajan omaavat tuotteet, jolloin saadaan selville myytävissä oleva saldo.
Tämän kaltaista saldolaskentaa käytettäessä saadaan päivän vähimmäisvalmistus laskettua seuraavasti:

Kuvio 18. Saldolaskenta

Käytännössä kuitenkin virheellisten pakkausten määrää on mahdotonta tietää tarkasti, joten useimmiten muuttuja Z (virheelliset pakkaukset) jätetään usein kokonaan huomiomatta tässä vaiheessa ja virheelliset pakkaukset otetaan huomioon huolehtimalla siitä, että muuttuja X (saldo) ei koskaan ole 0, vaan seuraavien myyntipäivien ennusteeseen suhteutettu positiivinen luku.

Kuvio 19. Vähimmäisvalmistuksen laskenta

3.4.4 Tuotantosuunnitelma

Tuotantosuunnitelman avulla kootaan tuotteen tietovarastoraportilta haetut ennustetiedot (Kuvion 20 alue D4:G4) sekä varaston lähtötilanne (Kuvion 19 solu D3) yhteen. Tämän jälkeen voidaan alueen D5:F5 solujen avulla suunnitella seuraavan päivän lähtötilanne eli varastosaldo.

Suunnittelualakävin määrittämiseen vaikuttavat lukuisat erilaiset tekijät, joista etenkin materiaalien ja raaka-aineiden saatavuus ovat keskeisissä rooleissa. Omakohtainen
kokemukseni sopivasta suunnitteluaikavälistä liha- alan yrityksessä on vähintään 3 viikkoa tulevaisuuteen johtuen raaka-aineiden saatavuuden rajallisuudesta ja työmäärän tasauksen tarpeesta.

Kuvio 20. Tuotantosuunnitelma

### 3.4.5 Tuotantolistat

Tuotantolistoille kootaan tuotteet sellaiseen muotoon, että kaikki informaatio on helposti työntekijän luettavissa. Lisäksi on otettava huomioon, että jos tuotanto aloittaa tuotteen valmistamisen ennusteeseen pohjautuen, voidaan ennustetta joutua korjamaan tuotannon aikana, jos ennuste on ollut pielessä.

Kuvio 21. Tuotantolista, jossa ennuste korjataan tarvittaessa lisätilauksella
4 TYÖN TARKASTELU

LÄHTEET


Karkea tuotannonsuunnittelu. [Verkkojulkaisu]. [Viitattu 24.3.2015]. Saatavana: 

Microsoft. Ei päiväystä. INDEKSI-funktio. [Verkkosivu]. [Viitattu 10.2.2015]. Saatavana:
https://support.office.com/fi-FI/Article/INDEKSIfunktio-a5dcf0dd-996d-40a4-a822-b56b061328bd

Microsoft. Ei päiväystä. Matriisikaavoihin liittyviä ohjeita ja esimerkkejä. [Verkkosivu]. [Viitattu 13.2.2015]. Saatavana:
https://support.office.com/fi-FI/article/Matriisikaavoihin-liit-tyvi%C3%A4-ohjeita-ja-esimerkkej%C3%A4-3be0c791-3f89-4644-a062-8e6e9ece523?CTT=3&CorrelationId=db4778a8-f63e-42aa-852f-ca61a2c9e33f&ui=fi-FI&rs=fi-FI&ad=FI

Microsoft. Ei päiväystä. PHAKU-funktio. [Verkkosivu]. [Viitattu 10.2.2015]. Saatavana:
https://support.office.com/fi-FI/Article/PHAKU-0bbc8083-26fe-4963-8ab8-93a18ad188a1

Microsoft. Ei päiväystä. SUMMA.JOS-funktio. [Verkkosivu]. [Viitattu 10.2.2015]. Saatavana:
https://support.office.com/fi-FI/article/SUMMAJOS-169b8c99-c05c-4483-a712-1697a653039b

Microsoft. Ei päiväystä. VASTINE-funktio. [Verkkosivu]. [Viitattu 10.2.2015]. Saatavana:
https://support.office.com/fi-FI/Article/VASTINE-e8dfdd45-c762-47d6-bf89-533f4a37673a

Microsoft. Ei päiväystä. VHAUKU-funktio. [Verkkosivu]. [Viitattu 10.2.2015]. Saatavana:
https://support.office.microsoft.com/fi-fi/article/VHAUKU-VHAUKU-funktio-a3034eec-b719-4ba3-bb65-e1ad662ed95f?CorrelationId=aef443ea-f520-4421-8371-fb8452dfb84&ui=fi-FI&rs=fi-FI&ad=FI