



CHALMERS

Analys inför förbättring av ett internt ordersystem

Examensarbete inom högskoleingenjörsprogrammet Mekatronik

LUDWIG OJALA

MIKAEL PAULSSON

Analys inför förbättring av ett internt ordersystem

Examensarbete inom högskoleingenjörsprogrammet Mekatronik

LUDWIG OJALA

MIKAEL PAULSSON



CHALMERS

Institutionen för Elektroteknik

CHALMERS TEKNISKA HÖGSKOLA

Göteborg, Sverige 2020

Analys inför förbättring av ett internt ordersystem

LUDWIG OJALA

MIKAEL PAULSSON

© LUDWIG OJALA, MIKAEL PAULSSON. 2020

Handledare: Robin Bandgren, Robert Chuanchuen, Acobia AB

Examinator: Veronica Olesen, Institutionen för Elektroteknik

Institutionen för Elektroteknik

Chalmers tekniska högskola

SE-412 96 Göteborg

Sverige

Telefon: +46 (0)31-772 10 00

Sammanfattning

Företaget som analyseras har ett ordersystem som har potential för förbättring och effektivisering. Beställningar kommer in och de överförs till en distributionsavdelning, distributionen omvandlar beställningarna till produktionslistor som innehåller produkter och ordningen de ska produceras i produktionsanläggningen. Produktionslistorna skapas med hjälp av olika Excel-dokument, skrivs sedan ut på papper och överförs manuellt till personalen i produktionen. När produktionen fått en produktionslista skapas tillverkningsordrar vid skärmar i produktionsanläggningen som bestämmer vad som ska tillverkas.

Flödet från beställning till färdig produkt har analyserats och personal har intervjuats för att få en god överblick hur företagets produktion fungerar i dag och hur det är möjligt att göra förbättringar.

Under intervjuerna har det framkommit att det inte är möjligt att automatisera hela vägen, rekommendationer för att lösa dessa problem har tagits fram.

En exempelwebbsida har tagits fram som utgår från det som kommit fram i analysen, webbsidan är till för att visa hur de olika delarna skulle kunna fungera tillsammans. Det är också möjligt att använda webbsidan som en bas för att bygga vidare på när de delar som behöver förbättras är klara.

Under analysen har det framkommit att personalen har ett behov av att kunna se exempelvis hur mycket som har tillverkats eller hur lång tid det är kvar tills en beställning är färdig. Därför har arbetet även lett till att visa hur det är möjligt att hantera data från en PLC till ett produktionsövervakningssystem som kallas för CitectSCADA som företaget använder.

Det här är starten för ett automatiskt ordersystem som kan utvecklas vidare och slutligen uppnå en orderhantering som inte kräver något eller lite manuellt arbete.

Abstract

The company that has been analysed has an order system that have potential for improvement. Orders come in and are then transferred to a distribution department, the department converts the orders to production lists containing the order in which different sort of the company's products should be produced in the production. The production lists are created using various Excel documents, then the production list is printed on paper and manually transferred to the staff in the production department. When the production gets the production list, they create production orders on screens at the production line that decides what is going to be produced.

The flow from order to finished products has been analysed and staff have been interviewed so that a good overview of how the production at the company are working and how it is possible to make improvements.

During the interviews it has become clear that it is not possible to automate the whole procedure, recommendations to solve these problems have been developed.

An example of a webpage has been created, it is based on what has emerged during the analysis. The webpage exists to show how the different parts may work together. It is also possible to use the webpage as a base to develop further when the parts that needs to be improved are ready.

During the analysis it has become clear that the staff for example has a need to see how much has been produced or how long time is left until an order is complete. Therefore, the work also has resulted to show how it is possible to handle data from a PLC system to a production monitoring system that is called CitectSCADA that the company are using.

This is the start of an implementation for an automatic ordering system that can be developed further and finally achieve an order system that does not require any or little manual work.

Förord

Den här analysen är ett examensarbete på högskolenivå som utförts av två studenter från mekatronikprogrammet. Arbetet har pågått under vårterminen och omfattar 15 högskolepoäng under institutionen för Elektroteknik på Chalmers tekniska högskola. Analysen har utförts åt företaget Acobia AB för en av deras kunder. Arbetet har till största del skett på Acobias kontor på Lindholmen i Göteborg.

Vi är tacksamma för vår tid hos Acobia och stödet vi fått från de anställda. Speciellt tack till Robin Bandgren och Robert Chuanchuen som väglett oss genom hela arbetet.

Vi vill även tacka den personal på företaget som analyserats som gett oss inblick och förståelse över deras anläggning samt hjälpt oss under arbetets gång.

Ludwig Ojala
Mikael Paulsson
Göteborg, juni 2020

Innehåll

1	Inledning	1
1.1	Bakgrund	1
1.2	Syfte.....	1
1.3	Mål.....	1
1.4	Avgränsningar	1
1.5	Precisering av frågeställning	2
2	Teoretiskt ramverk	3
2.1	Databaser	3
2.2	SQL.....	3
2.3	HTML	3
2.4	CSS	4
2.5	Javascript	5
2.6	CitectSCADA	5
2.7	PLC.....	5
3	Metod	7
3.1	Upplägg	7
3.2	Förstudie	7
3.3	Intervjuer	7
3.4	Utvärdering.....	7
4	Förstudie	8
4.1	Beställningar.....	8
4.2	Behovslista.....	9
4.3	Körordning.....	9
4.4	Produktionslista	10
4.5	Tillverkningsorder	11
4.6	Produktionen.....	12
4.7	Kravspecifikation.....	13
5	Webb	14
5.1	Kravställning för webbsida.....	14
5.2	Exempel för webbsida	15
5.2.1	Uppbyggnad	15
5.2.2	Informationssida.....	16
5.2.3	Orderhantering	16

5.2.4	Körordning	17
5.2.5	Tillverkningsorder	19
5.2.6	Historik och datum	19
5.3	Kommunikation mellan webb och databas.....	20
6	Återkoppling	21
6.1	Simulerad återkoppling.....	21
6.2	Simuleringsexempel	21
6.3	Hårdvara och inkoppling	22
7	Resultat	24
8	Diskussion.....	25
9	Vidareutveckling.....	26
10	Slutsats	27
11	Källförteckning.....	28
	Bilagor	29

1 Inledning

Detta kapitel kommer ge en bild av varför denna analys gjorts och beskriver framtagningen av frågeställningen, syfte och målsättningen.

1.1 Bakgrund

Projektet skapades genom ett samarbete mellan Acobia AB och deras kund. Projektet är utformat som en analys som går ut på att ta fram ett lösningsförslag på hur det analyserade företagets interna ordersystem skulle kunna förbättras för att underlätta och modernisera arbetssättet. Dagens system innehåller sköra delar som kan framkalla missförstånd, felläsning och slutligen förlorad produktion. Beställningar kommer in och med manuell räkning skapas en Excel-lista med en körordning. Denna lista levereras till produktionen som skapar tillverkningsordrar online i ett webbgränssnitt som är kopplat till maskinerna i produktionen som kan starta tillverkningen.

1.2 Syfte

Uppdraget är att analysera och ta fram en möjlig lösning för att ersätta Excel-dokumentet och den manuella hanteringen av orderarna. Resultatet skall vara ett underlag för att kunna påbörja moderniseringen av den interna orderhanteringen på ett sätt som anpassas efter de olika avdelningarnas önskemål.

1.3 Mål

Målet är att skapa en mall för hur systemen för orderhanteringen skall se ut och vilka funktioner de kommer kräva för att kunna implementeras hos företaget.

1.4 Avgränsningar

Arbetet och analysen sker på en teoretisk nivå vilket medför att det inte kommer skapas ett färdigt system och resultatet kommer baseras mer på intervjuer. En del exempel kommer skapas för att visa hur delmoment och design kan se ut och fungera.

1.5 Precisering av frågeställning

Hur skulle en automatisering av företagets ordersystem kunna utformas? Detta är hela projektets grundfråga.

I detta finns flera underliggande frågor:

- Hur ska webbgränssnittet utformas?
- Hur kan en webbsida kommunicera med SQL databaser i MSSQL?
- Hur kan feedback från produktionen fungera?
- Hur kommer orderarna in?
- Vilka moment kan ersättas?

2 Teoretiskt ramverk

Här beskrivs övergripande om olika tekniska delar som tas upp i rapporten och används i analysen.

2.1 Databaser

Med hjälp av en databas kan man spara och organisera data på ett strukturerat sätt. Databasen gör det sedan lätt och effektivt att hitta och hämta ut just den data man vill använda för tillfället. En databas är ett register som innehåller ett antal olika fält, fälten är uppstrukturerade på så sätt att det bildar en tabell med rader och kolumner där all data sparas [1, s. 8-9].

2.2 SQL

En databas är platsen för att spara all data, mjukvaran som en programmerare använder för att hantera datan kallas för en databashanterare eller DataBase Management System (DBMS) på engelska [1, s.8].

Programspråket som används är Structured Query Language (SQL) och den har tre funktioner. Skapa och strukturera informationen och databasen, hämta ut informationen och att hantera säkerheten. Med säkerhet menas exempelvis vem som ska ha tillgång till datan i databasen genom att använda en inloggning för användaren [1, s.11].

2.3 HTML

HTML står för Hyper Text Markup Language och används som grundspråk till webbsidor. Det är ett eget språk som med hjälp av taggar skapar olika element och uppbyggnaden av en webbsida [2], se figur 2.1 för att se olika element och taggar. Ett HTML program skrivs i filformatet .html som sedan läses av genom en webbläsare. HTML är inte ett programmeringsspråk utan ett märkspråk, se namnet "Hyper Text Markup Language", detta betyder att det är för att skapa upplägget på sidan med länkar och tabeller men de kan inte sköta funktioner som då de hade varit ett programmeringsspråk [3, s.1].

```

<!DOCTYPE html>
<html>
  <head>
    <title></title>
    <meta charset="utf-8" />
    <link rel="stylesheet" href="style.css">
  </head>

  <body>
    <h1>
      
      Välkommen till ordersystem
    </h1>

    <h2>Länk till ordersystem</h2>

    <button type="button" onclick="window.location.href =
'http://localhost:5000';">Se ordrar</button>

  </body>
</html>

```

Figur 2.1. Bild på HTML-kod. (Författarnas egen bild)

2.4 CSS

CSS är förkortning av Cascading Style Sheets. Det är ett språk för att beskriva layouten av en webbsida. Det kan ingå i HTML filen eller vara en extern fil som länkas till HTML filen. En design kan användas till flera element i HTML programmet som skall ha samma design [4]. Se figur 2.2 för exempel på kod som ger elementen i HTML-koden den angivna designen.

```

@charset "UTF-8";

h1{
  background-color: rgb(191, 49, 27);
  margin: 20px;
  padding: 20px;
  text-align: left;
  color: rgb(243, 210, 61);
}

.rader {
  background-color: rgb(191, 49, 27);
  color: rgb(243, 210, 61);
  margin: 20px;
  padding: 20px;
}

img {
  float:left;
  width:150px;
  height:100px;
  margin:-22px;
}

table{
  width:100%;
  border:1px solid black;
}

th, td {
  width:50px;
  border:1px solid black;
}

```

Figur 2.2. Bild på CSS-kod. (Författarnas egen bild)

2.5 Javascript

Javascript används för att förändra HTML-innehåll och för att sköta funktioner som hanteras av webbsidor. Javascript är inte ett programmeringsspråk som grundar sig i Java som namnet antyder, det har bara fått sitt namn från Java på grund av att det var populärt när Javascript skapades [5, s.3].

Javascript är ett programmeringsspråk som är interpreterande, vilket betyder att koden skickas vidare till ett översättande program som översätter koden till maskinkod så att datorn ska kunna förstå och utföra koden. Översättningen är inbyggd i webbläsaren och sker alltså när koden körs. Det finns även andra program som kan hantera Javascript, Node.js är ett exempel. En del andra programmeringsspråk kompilerar istället koden och ersätter den med maskinkod i förväg. [5, s.2]

Javascript används mot både HTML och CSS för att göra beräkningar och manipulera data. Många webbsidor använder sig ofta av alla tre språk för att skapa den optimala hemsidan. Javascript är användbart både för server- och klientsidan [6, s2].

2.6 CitectSCADA

CitectSCADA används för övervakning och styrning av produktion och då framförallt för PLC-system. Programmeringsspråket som används för att programmera CitectSCADA kallas för Cicode. Det finns olika programvaror för att programmera Cicode, ett av dem är mjukvaran Citect Studio. Citect Studio har många olika funktioner, det används för att programmera Cicode men har även funktioner som att skapa en grafisk sida för processövervakningen. [7]

2.7 PLC

Programmable logic controller, eller programmerbart styrsystem på svenska, är ett system för att automatisera styrningen av exempelvis en produktionsanläggning i en fabrik. Dess föregångare bestod av bland annat reläer, räknare och klockor men krävde onödigt krångliga el-installationer och arbete om något behövdes ändras på så det togs fram en ersättare för det äldre systemet vilket blev PLC. För att bli av med problem som fanns tidigare består PLC av en programmerbar processor som det kopplas delar till, det gör att alla möjliga olika funktioner som tidigare behövde kopplas ihop istället sköts av processorn. Processorn får olika sorters signaler av sensorer och switchar, signalerna behandlas i ett av användaren uppbyggt program som processorn styrs av och den skickar sedan ut signaler som styr exempelvis pumpar och motorer utifrån det program användaren har ställt in processorn att utföra. Till PLC:n går det sedan att koppla datorer för att övervaka systemen med olika sorters programvaror, exempelvis CitectSCADA. [8, s.24-37]

PLC-systemen tillverkas av många olika företag med olika lösningar, Mitsubishi, ABB och Siemens är exempel på några av de större tillverkarna

För att programmera en PLC används flera olika typer av programmeringsspråk vilka lämpar sig för olika användare och för olika situationer. De programmeringsspråk som främst används är Function Block Diagram (FBD), Instruction List (IL), Ladder Diagram (LD), Sequential Function Chart (SFC) och Structured Text (ST). FBD, LD och SFC är i form av grafisk programmering, IL och ST är textbaserade exempelvis likt HTML eller C. [8, s.200].

FBD fungerar på så sätt att man använder sig av olika funktionsblock som sedan kopplas ihop med varandra. Det används även en del Booleska funktioner mellan funktionsblocken för att kunna manipulera indata [8, s.206].

3 Metod

Metodavsnittet ger en övergripande bild av arbetets upplägg.

3.1 Upplägg

Arbetet har utförts på plats hos Acobia, genom deras stora programvaruutbud och förkunskaper inom dessa blev möjligheten för ett smidigt arbete större. En förstudie har gjorts genom att hålla intervjuer med personal på företaget som analyseras för att skapa en bild av dagens situation och vad som skulle förändras. Arbetet mot en lämplig lösning utfördes genom att utvärdera idéer och förslag som togs fram under arbetets gång. Exempel med lösningar på de olika delmomenten togs fram och presenterades för personalen för att få feedback och göra utvärderingen möjlig.

3.2 Förstudie

För att skapa en bild av hur det fungerar idag och hur det kan förbättras gjordes en förstudie som visade dagens process och vilka problem som lyses upp.

3.3 Intervjuer

För att ta reda på hur det fungerar idag och för att skapa en kravspecifikation som uppfyller målen besöktes företaget och intervjuer hölls med de berörda avdelningarna, från början gav personalen sin bild av vad de önskar och varför. När kravspecifikationen var färdig kunde frågeställningar skapas och vara underlag för de löpande intervjuerna på plats, via teams samt genom mejlkontakt.

3.4 Utvärdering

Eftersom målet med analysen är att ta fram ett lämpligt lösningsförslag måste flera idéer bearbetas och jämföras för att komma fram till den mest passande lösningen. Slutmålet som togs fram kan uppnås på flera olika sätt då de innehåller många delmoment som skall fungera tillsammans. Genom att arbeta fram exempel och idéer för möjliga arbetssätt kunde de presenteras för personalen och anpassas utifrån deras önskemål tillsammans med kravspecifikationen.

4 Förstudie

Företaget som analyseras arbetar på ett unikt sätt för att uppnå sina försäljningsmål. Det är många tider och regler att ta hänsyn till för att få ut produkterna till butikerna runt om i landet men även utanför Sveriges gränser.

I detta avsnitt beskrivs hur den interna orderlogistiken fungerar och en kravspecifikation tas fram som beskriver önskat slutresultat.

4.1 Beställningar

Områdessäljare runt om i landet har ansvar för beställningar till närliggande butiker. De lägger ordrar via ett elektroniskt system som heter Vudiax. Beställningarna innehåller data om hur mycket av varje sort och till vilka butiker produkterna ska skickas. Dessa beställningar för ett område sammanställs till gemensamma ordrar för en distributionsort som sedan skickas till företaget genom Vudiax. Dessa olika distributionsorter är bundna till packomgångar som avgör när produkten skall vara klar för att anpassas till lastbilarnas avgångstider. I Vudiax genereras en behovslista som visar hur mycket som behöver produceras och vilken tid som en produkt behöver vara färdig för att kunna packas på en viss lastbil.

En stor del av företagets beställningar kommer inte från Sverige utan är exportbeställningar. Eftersom företaget producerar färskvaror så är det framförallt till grannländer som Danmark och Norge som exporten sker, men företaget exporterar även till länder som Frankrike och USA. När exporten sker till USA är det framförallt varor med längre hållbarhet som beställts.

Till skillnad från nationella beställningar görs exportbeställningarna inte genom Vudiax då Vudiax är till för säljarna inom Sverige och inte används till kunder utanför landets gränser. Exportbeställningar från olika länder kommer in via olika system men oftast via mejl och behandlas därefter på olika sätt. Dessa beställningar skrivs in i Excel-tabeller för aktuell dag och tillverkningslinje.

En annan del av beställningarna är STH som står för storhushållsbeställningar och det kallas beställningar från större företag som lägger stora ordrar för att använda produkterna till sina egna varor. Man har valt att särskilja dessa ordrar från de vanliga orderarna från butikerna, dels för att dessa skiljer sig en del och även för att ha större koll på hur mycket som beställs av just de här kunderna. De här kunderna ligger inte inne i det vanliga Vudiax-systemet så beställningarna kommer inte heller den vägen, utan beställningar kommer även här via mejl. Storhushåll kan exempelvis vara en flygplats där man köper in stora mängder för att producera sina egna produkter som sedan serveras som mat på flygplanen under flygningarna. Ett annat exempel är Ikea som gör väldigt stora beställningar av vissa produkter som sedan säljs i Ikeas varuhus.

Även lagersaldon kontrolleras och används vid beräkningen av vad som skall produceras. Ibland överproduceras vissa produkter för att det finns tid över på produktionsdygnet. Detta görs för att underlätta kommande produktionsdygn och få ut alla sorter då beställningarna mellan dygn varierar. När ett nytt produktionsdygn skall planeras så läses lagersaldon av och räknas av från beställningar för att bara producera nödvändig mängd.

4.2 Behovslista

Eftersom distributionsorterna är sammankopplade med avgångstider så blir det lätt att sortera upp en behovslista som visar när en viss mängd av en viss sort skall vara färdigproducerad. I behovslistan samlas alla beställningar från Vudiax som skall produceras under ett produktionsdygn. En rad för varje sort och kolumnerna är uppdelade så att de visar hur mycket som behöver vara färdigt vid vissa tidpunkter för att det ska hinna med olika packomgångar

De olika distributionsorterna har olika packomgångar vilket avgör vilken kolumn de hamnar i. Packomgångar är uppdelat i tiondelar så exempelvis kan tre distributionsorter hamna i samma kolumn men ha olika packomgångar så som 3.1, 3.2 och 3.3. Beroende på vilken tid lastbilen ska köra så kan de även hamna i olika tidskolumner, siffrorna i packomgången står alltså inte för vilken tid packomgången ska avgå. Denna lista ger en tydlig överblick av alla beställningar från Vudiax som skall vara med i planeringen av körordningen. Ett exempel på en behovslista som företaget använder visas i figur 4.1.

		ONSDAG			TISDAG				EXTR.	DEL	NATT TISDAG - ONSDAG											EXTR.	LAGER	DEL	TOT
			12	16	18	22	ORD.	NATT	SUM	1	2	3	4	5	6	7	9	11	ORD.	JUST.	SUM	SUM			
20:AN	621								0												0	0			
	862								0												0	0			
35/37	604								0												0	0			
	605								0												0	0			
	610								0												0	0			
	611								0												0	0			
	612								0												0	0			
39:AN	177			35	88		17		140	128		3	19	26	8		17		4		205	345			
	833			19	11		15	28	73							10	10		1		21	94			
50:AN	173			187	105		10	425	727						59	34		1		94	821				
	865			202	140		33		375	443		249	157	197	89	28		13		1176	1551				
51:AN	811			88	71		81	590	830							23		1		24	854				
	847			165	165		52	370	752						220	44		1		265	1017				
	849			128	101		20		249	301		20	56	85	41	79		12		594	843				
	870			272	283		54	740	1349						111	128		9		248	1597				
	871								0											0	0				
52:AN									0											0	0				
									0											0	0				
53:AN	141			219	60		27		306	331		39	173	182	69	49		4		847	1153				
									0											0	0				
60:AN	107			528	493		118	644	1783	1119		104	256	315	139	212		52	-731	1466	3249				
	863			187	114		106	449	856						53	49		2		104	960				
61:AN	143			594	493		67		1154	1081		66	257	333	153	405		24		2319	3473				
	864			118	86		28	168	400						95	47		1		143	543				
	156								0											0	0				
62:AN	805			231	275		37		543	562		11	47	57	29	49		4		759	1302				
									0											0	0				

Figur 4.1. Skärmdump av en behovslista. (Författarnas egen bild)

4.3 Körordning

Behovslistan från Vudiax, STH och exportbeställningar från mejl samt lagersaldon sammanställs med handpåläggning för att skapa en körordning utifrån vilka avgångstider de har. De

delmängder av sorter som skall vara klara till en viss tid planeras in i en ordning som garanterar att alla sorters beställningar är färdiga till önskad tid för att hinna med den planerade leveransen. Allt av en sort kan ofta inte köras direkt då andra sorter måste prioriteras för att färdigställas till sina respektive avgångstider. För detta moment krävs mycket tid och planering som baseras på variabler med ledtider, produktionshastigheter och avgångstider för lastbilarna. Detta moment sker i en Excel-fil och det kan ses som ett verktyg för att skapa det som sedan blir produktionslistan. I figur 4.2 visas ett exempel på en körordning från Excel-filen. För att få ut färdiga produkter till 21:00 måste första satsen vara färdig att läggas i maskinen vid tidpunkten som med ledtiden blir färdig 21:00. Starttiden är alltså önskad färdig tid minus ledtiden.

TISDAG	849	1104	1896	276	847	3238	3239	3186
Körordning linje 51								
DATUM								
TILLGÄNGLIG PROD.TID	09,35							
STARTTID	17,45	20,25	20,25	20,45	21,41	23,28	23,48	23,48
PRODUKT								
ANT. KART/PALL	40	40	36	36	40	28	32	36
ARTIKELNUMMER	849	1104	1896	276	847	3238	3239	3186
	80	80	80	80	72	72	72	72
ENH./MINUT.	48	48	48	48	55	55	55	55
KART./TIMME	420	420	420	420	425	425	425	425
MÄNGD ENH. ATT PRODUCERA	1117		144	180	762	140		1
UPPEHÅLLSTID TILL NÄSTA				0,30				0,30
PRODUKTIONSTID Produkt	2,40	0,00	0,21	0,26	1,48	0,20	0,00	0,00
ÅTERSTÅENDE Produktionstid	6,55	0,00	6,35	5,39	3,52	3,32	0,00	3,02

Figur 4.2. Skärmdump av en körordning. (Författarnas egen bild)

4.4 Produktionslista

Produktionslistan innehåller olika typer av information som produktionen behöver för att kunna tillverka det som kunderna har beställt, se figur 4.3 för ett exempel av en produktionslista. Informationen i produktionslistan är vilken produkt som ska tillverkas, vilket artikelnummer den har, hur många enheter som ska produceras för varje produkt och totalt hur många kartonger som ska tillverkas. En annan viktig information man får ur produktionslistan är vid vilka tider som varje produkt ska packas i påse, detta anges genom start- och slutpaketeringstider. Produktionslinjerna körs med en viss hastighet, vilken även anges i produktionslistan genom antal kartonger per timme. Utifrån den här informationen kan sedan produktionen beräkna när man ska börja producera och när de olika produkterna ska startas för att de ska vara klara till tiderna för packningen och lastning på lastbilarna.

Produktionslistan skrivs ut på papper och lämnas över två gånger per dag, först en preliminär och senare en slutgiltig efter klockan 18. Många produkter behöver förberedas flera timmar innan de kan börja köras i produktionslinjerna, därför används den preliminära produktionslistan så man vet ungefär hur mycket som kommer behöva produceras för att hinna förbereda i tid. Vissa produkter tar så lång tid att förbereda att man inte har hunnit få den preliminära listan,

för dessa produkter måste produktionen ringa till distributionen och ta reda på det istället. Den slutgiltiga produktionslistan är till för att alla butiker ska hinna göra kompletteringar till sina beställningar innan produktionsordningen är fastställd.

När produktionslistan är fastställd och produktionen har startat igång kan det dyka upp problem, det kan finnas för lite av en råvara eller en maskin kan sluta fungera. I det fallet behöver produktionslistan ändras, en del av produktionstiden försvinner och vissa produkter kan behöva prioriteras och produktionslistan ordning uppdateras.

PRODUKTIONSLISTA LINJE 51												
TISDAG												
			Utbyte									Kapacitet
Artnr	Sort			Avrund	Kartonger	Timmar	Start uppsl	Start pkt	Slut pkt	Pallar	Kart	Kart/tim
849		13,96	80		1117	2,40	17,45	21,00	23,40	27	37	420
1104					0							
1896		1,80	80		144	0,21	20,25	23,40	00,00	4	0	420
276		2,25	80		180	0,26	20,45	00,00	00,26	5	0	420
847		10,58	72		762	1,48	21,41	00,56	02,43	19	2	425
3238		1,94	72		140	0,20	23,28	02,43	03,03	5	0	425
3239					0							
3186		0,01	72		1	0,00	23,48	03,03	03,03	0	1	425

Figur 4.3. Skärmdump av en produktionslista. (Författarnas egen bild)

4.5 Tillverkningsorder

En av produktionslinjerna har idag ett eget system som är uppkopplat och körs mot en databas samt mot produktionens system CitectSCADA. Det är en webbsida som hanterar tillverkningsordrar för att se status samt sköta uppstarten av produktionen.

En tillverkningsorder består av flera olika delar. Först finns ett order-id, därefter finns information om vilken tapp och linje som tillverkningsordern gäller. Tillverkningsordern innehåller även statusinformation, det kan exempelvis vara att ordern inte är komplett vilket betyder att ordern till exempel saknar ett kopplat recept. Annan information som står med är önskat antal och producerat antal, helt enkelt hur mycket som har beställts och vad som är färdigproducerat. Slutligen finns det en beskrivning, där står vilken produkt eller annan information som produktionen behöver veta för just den tillverkningsordern.

Tillverkningsordern skulle kunna ses som en förenklad version av produktionslistan som bara innehåller det som maskinen och operatören behöver veta för att starta tillverkningen. Produktionslistan som är planeringen av körordningen med dess starttider är uppdelad på varje order. Oftast så läggs fler ordrar som till exempel ska till olika länder men är av samma sort efter varandra i produktionslistan. Då dessa produkter är samma sort och har samma innehåll samt körs efter varandra så kan tillverkningsordern som är för tillverkningen komprimeras till

en gemensam tillverkningsorder som visar hur många satser som krävs för att köra de ordrar i produktionslistans totala antal kartonger.

4.6 Produktionen

Produktionen sker under natten på grund av att man tillverkar färskvaror som helst ska vara i butikerna redan under morgonen. Om en produkt är färdigt för att packas tre timmar innan dagen är slut får förpackningsdatum anges till nästa dag [9]. Normalt sett startar därför produktionen så att första färdiga produkten packas i påse precis efter klockan 21.

När en produktionslista levereras skriver operatören in den som tillverkningsordrar på en webbsida, som just nu bara finns sammankopplad med en produktionstapp (tapp 5). En produktionstapp är en maskin där man får ut råvarorna i produktionsanläggningen. Operatören kopplar ihop tillverkningsordern med ett recept för varje produkt, sedan görs en koll med produktionssystemet om alla råvaror finns och produktionen startas igång.

För de övriga tapparna används inte systemet med en webbsida, här knappar istället operatören in hur mycket som behöver tillverkas direkt till produktionsmaskinerna. Tanken är att alla tappar och produktionslinjer ska få systemet med webbsidan i framtiden.

Tillverkningsordern skickas in till en databas, här läser CitectSCADA kontinuerligt av om det kommit in någon ny order som också markerats att den ska köras. Citect skickar sedan vidare data till PLC-systemet som startar igång produktionsanläggningen.

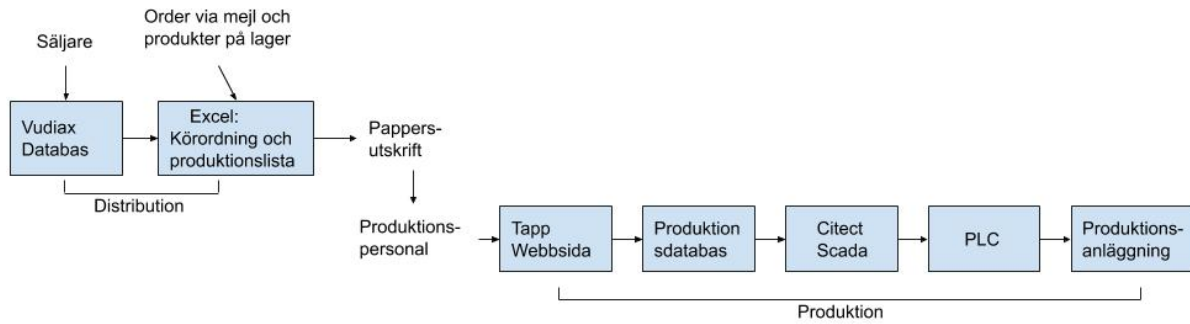
Produktionen är uppdelad i flera olika delar, den första delen är råvarulagret. Där blandas råvarorna som med hjälp av vågar väger upp hur mycket av varje del som ska skickas vidare till de olika tapparna vid produktionslinjerna.

Från tapparna tar man ut råvarorna, tillsätter övriga ingredienser och börjar sedan bearbeta dessa. Ingrediensblandningen går sedan igenom flera olika produktionssteg för att till slut komma ut som en färdig produkt.

När produktionen är klar skickas produkten vidare till paketeringen som befinner sig i slutet av varje produktionslinje. De olika produkterna packas i påse, påsarna packas i lådor och därefter skickas produkten till en lastcentral där lådorna packas på pall. Pallarna ställs sedan på olika lastbilar som transporterar ut den färdigproducerade produkten till olika distributionscentraler. Från dessa centraler hämtar sedan försäljarna upp produkten och kör ut det till butikerna.

Varje lastbil har en avgångstid som måste hållas för att produkten ska hinna fram till en viss butik. Är inte produkten färdigproducerad och packad vid den tiden måste lastbilen åka iväg utan att få med sig fullständiga beställningar. Därför är det viktigt för produktionen att kunna prioritera vissa sorter vid vissa tidpunkter för att allt ska hinna produceras i tid till varje lastbil.

Flödesschema



Figur 4.4. Flödesschema från säljare till produktionsanläggning hos företaget. (Författarnas egen bild)

4.7 Kravspecifikation

En kravspecifikation togs fram utifrån slutmålet med projektet tillsammans med information från personal på företaget som jobbar med de berörda momenten.

- Pappershanteringen skall bort.
- Tillverkningsorder genereras automatiskt.
- Förenkla uppdatering av tillverkningsordrar och produktionslistor under produktionens gång.
- Sammankopplat mellan körordning till tillverkningsorder.
- Ersätta Excel-filen med en webbsida.
- Generera körordning automatiskt.
- Feedback från produktionen.

5 Webb

För att hantera planeringen med ordrar på ett nytt och smidigare sätt så är en webbsida som sköter de funktioner som idag görs i Excel ett exempel på hur systemet kan moderniseras.

I detta avsnitt presenteras hur det är tänkt att en webbsida kan fungera.

5.1 Kravställning för webbsida

Utifrån förstudien som gjorts har krav och funktioner för en webbsida tagits fram.

Arbets sättet kommer vara som innan, men alla delar som krävs för planering och orderhantering samlas under en webbsida. På så sätt blir det mer lättåtkomligt och effektivare för planeraren i distributionen. Fördelen med webbsidan är även att behovet av att skriva ut produktionslistor på papper försvinner och istället kommer produktionen kunna se behovslistor och tillverkningsordrar direkt på en skärm vid tapparna.

I ett första steg kommer det vara bra att samla alla inkommande beställningar på en sida för att enklare få överblick. Eftersom all data inte finns tillgänglig i databaser kommer distributionen för vissa delar som exempelvis exporten manuellt behöva överföra data till den nya webbsidan, på ett liknande sätt som man gör idag fast då är det till Excel-filerna. Vudiaxdatan kan däremot hämtas direkt till webbsidan genom att webbsidan ska kunna prata med databasen.

Under en annan sida ska det finnas funktioner för att se och skapa produktionslistor. Här ska det finnas en sida som innehåller en körordning, med samma funktioner som i dagens Excel-dokument. Här behöver det även finnas en sida för att se färdiga produktionslistor. Produktionslistan behöver finnas tillgänglig för förpackningsdelen av produktionen eftersom beställningar med olika förpackningar men samma innehåll slås ihop till samma tillverkningsorder. Förpackningsdelen behöver kunna se exempelvis hur många påsar av en sort som ska produceras.

En annan del som behöver finnas på webbsidan är en sida för tillverkningsordrar. Här läggs färdiga tillverkningsordrar som ska skapas utifrån körordningen eller produktionslistan. Den här sidan behöver innehålla verktyg för att redigera, ta bort och lägga till tillverkningsordrar för att distributionen ska ha möjlighet att rätta till något som eventuellt har blivit fel och öka flexibiliteten. Något annat som behöver finnas och som också är med i kravspecifikationen är möjligheten för distributionen att först granska alla tillverkningsordrar för att sedan godkänna dessa. Därefter för distributionen över tillverkningsordrar till produktionen. Här är det bra om det finns möjlighet att antingen skicka hela listan med tillverkningsordrar på en gång men även att individuellt kunna skicka enstaka tillverkningsordrar. Det kan även vara bra att det kommer upp ett meddelande att inga tillverkningsordrar har valts om man trycker på knappen för att skicka tillverkningsordrar och ingen tillverkningsorder är markerad.

Möjligheten att kunna skapa tillverkningsordrar utan att skapa dem via körordningen är en annan viktig funktion som behöver vara med för produktionens skull. Vissa sorter kräver lång tid av förberedelse innan det är möjligt att börja producera. I det här fallet är det bra om distributionen kan lägga upp tillverkningsordrar åt produktionen så tidigt som möjligt, exempelvis innan körordning och produktionslistan är färdiga.

Många av funktionerna ovan är till för att man ska slippa springa med papperslappar och ringa varandra för att få reda på allt som ska produceras. Att ta bort telefonsamtalen helt kommer vara svårt, men det går att få bort många av dem vilket leder till ett effektivare arbete som sparar tid. För att det hela ska fungera smidigt är det bra om det finns en funktion som gör att produktionen blir uppmärksam på att listan med tillverkningsordrar har uppdaterats. Det kan exempelvis vara ett popup-fönster på webbsidan som produktionen måste godkänna för att ta bort och där det står att listan är uppdaterad.

Om listan för tillverkningsordrar ska vara dynamisk och uppdateras behöver tillverkningsordrarna även i fortsättningen visa status. Exempelvis kan statusen vara att det står att den körs, för att på så sätt göra att det inte av misstag går att köra en tillverkningsorder två gånger. Om tillverkningsorderlistan uppdateras så kan en tillverkningsorder hamna längre ner än innan och då kan en operatör lätt missa detta och köra samma tillverkningsorder igen.

Eftersom olika delar i företagets tillverkning behöver ha tillgång till olika delar av webbsidan finns det ett behov av att använda sig av inloggningar. Produktionen ska inte ha tillgång till alla delar, för dem räcker det att ha tillgång till en sida där det finns behovslista, tillverkningsordrar och historik. Eventuellt kan det även vara bra att kunna redigera tillverkningsordrarna, ifall man till exempel behöver justera storleken på ordern. Planeraren i distributionen behöver ha tillgång till alla delar eftersom alla delar påverkar skapandet av körordning och tillverkningsordrar. Andra personer i distributionen som exempelvis har hand om exportordrar kan ha tillgång till att lägga in ordrar i orderlistorna och historik för orderna.

5.2 Exempel för webbsida

Ett exempel på ett utseende för en webbsida har skapats. Tanken med exempelsidan är att visa upp hur de olika delarna ska fungera tillsammans, men den är även till för att få tillbaka feedback på om något skulle behöva ändras.

5.2.1 Uppbyggnad

Exempelsidans innehåll är framförallt kosmetiskt. Uppdraget för företaget har aldrig varit tänkt att ge en färdig produkt, därför är inte de flesta funktionerna implementerade på webbsidan. Den pratar inte heller med någon databas för att hämta in information om till exempel beställningar.

Exempelsidan bygger på html, CSS och Javascript. HTML är själva grunden för allt innehåll och utseende. CSS hanterar utseendet på många av funktionerna, exempelvis knappar, textstorlek eller var på sidan något ska placeras. Javascript hanterar funktioner så som att knappar går att trycka på, knapparna är främst till för att ta sig till någon annan del på sidan.

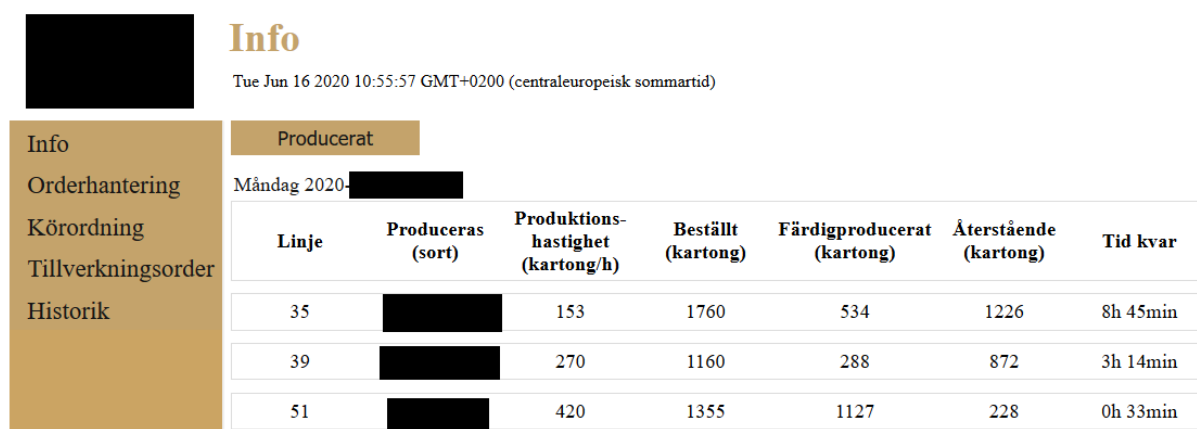
Utseendet bygger på funktionerna som har beskrivits under kapitlet om webbsidan. Vissa delar bygger även på den befintliga webbsidan som redan nu finns på en skärm vid tapp 5 i produktionen, men den har modifierats en del för att kunna hantera tillverkningsordrar för alla linjer i produktionen istället för bara en. Andra delar som exempelvis hantering av inkommande

ordrar ser precis likadant ut som det gör i de olika Excel-filerna som personalen använder sig av idag, men istället finns de på en webbsida.

Exempelsidan är uppbyggd av en sidebar på vänster sida som innehåller en meny, sedan visas webbsidans innehåll till höger, för att se hur sidebaren är skapad, se bilaga 1. Högst upp finns en text som beskriver vilken sida man har valt när man klickat på en knapp i menyn. Under menyn finns en text som innehåller dagens datum och vad klockan är för tillfället för att detta ska vara lättåtkomligt. Webbsidan innehåller fem sidor eller avdelningar, info, orderhantering, körordning, tillverkningsorder och historik. Under varje sida finns undersidor, undersidorna behövs för att all information och alla funktioner inte ska läggas på en sida vilket lätt skulle göra att det blir oordning.

5.2.2 Informationssida

Infosidan är en sida för information från produktionsanläggningen som ska uppdateras i realtid, se figur 5.1 för hur infosidan ser ut. Här visas vilken linje som informationen kommer ifrån, vilken sort som produceras just nu och produktionshastigheten. Det visas även hur mycket som beställts, färdigproducerat och hur mycket som är kvar att producera, dessa visas i antal kartonger. Antal kartonger är valt istället för antal satser eftersom det är distributionen som har behov av att se information från produktionsanläggningen. Slutligen visas även hur lång tid det är kvar för den sorten som produceras tills att alla kartonger är klara.



Linje	Produceras (sort)	Produktionshastighet (kartong/h)	Beställt (kartong)	Färdigproducerat (kartong)	Återstående (kartong)	Tid kvar
35		153	1760	534	1226	8h 45min
39		270	1160	288	872	3h 14min
51		420	1355	1127	228	0h 33min

Figur 5.1. Infosida från exempelwebb. (Författarnas egen bild)

5.2.3 Orderhantering

Nästa sida i menyn är till för orderhantering, se figur 5.2 och 5.3 för hur orderhanteringssidan ser ut. På denna sida kan planeraren hämta information om hur mycket som beställts. Informationen behövs sedan när körordningen och produktionslistan ska skapas. Det finns fem olika flikar under orderhantering, inrikes, export, STH, övrigt och lager. Här kan man välja vilken del av de inkommande beställningarna som man vill ha data från. Under inrikes finns en knapp för att importera behovslista, den ska vara kopplad direkt mot Vudiax för att hämta in

behovslistan till webbsidan på samma sätt som det går att göra i Excel-filen som används idag. Här finns även en sida för att se vilka produkter och hur mycket av dem som finns på lager, denna sida ska uppdateras automatiskt genom att vara kopplad till ett lagersystem. De övriga sidorna är tänkta att det manuellt förs in data i listorna på samma sätt som det görs i Excel-filer idag. Det är bra om listorna har möjlighet att sorteras beroende på vad användaren vill se för tillfället.

Info

Orderhantering

Körordning

Tillverkningsorder

Historik

Orderhantering

Tue May 19 2020 13:46:48 GMT+0200 (Central European Summer Time)

Inrikes

Export

STH

Övrigt

Lager

Inrikes

Ändra datum

Måndag 2020

Importera Behovslista

Linje	Art.no	Artikel	12	16	18	22	Extra-ord.	Natt	Del-summa	1	2	3	4	5	6	7	9	11	Extra-ord.	Lager-just.	Del-summa	Total-summa
39	177		0	80	42	104	139	0	365	342	0	11	76	43	32	1	17	0	1	0	523	888
39	833		0	56	27	69	25	189	366	0	0	0	0	0	34	1	8	0	1	0	44	110
51	811		0	117	81	155	51	1020	1424	0	0	0	0	0	0	33	22	0	1	0	56	1480
51	847		0	503	0	0	0	1509	2012	0	0	0	0	0	323	0	56	0	1	0	380	2392
51	849		0	257	85	195	338	0	875	640	0	21	127	107	100	4	64	0	9	0	1072	1947
51	870		0	444	0	0	0	1934	2378	0	0	0	0	0	168	11	129	0	1	0	309	2687

Figur 5.2. Orderhanteringssida från exempelwebb. (Författarnas egen bild)

Orderhantering

Wed May 20 2020 14:02:55 GMT+0200 (Central European Summer Time)

Info

Orderhantering

Körordning

Tillverkningsorder

Historik

Inrikes

Export

STH

Övrigt

Lager

Export

Ändra datum

Måndag 2020

Linje	Art.no	Artikel	Beställare ▼	Antal
61	1072		Danmark	398
51	1104		Danmark	75
52	1160		Finland	540
51	3186		Finland	368

Figur 5.3. Orderhanteringssida från exempelwebb. (Författarnas egen bild)

5.2.4 Körordning

Under sidan för körordning finns ett program/funktion för att lägga upp körordningar och sedan en knapp som genererar en produktionslista utifrån körordningen, se figur 5.4. De celler som ska vara redigerbara i körordningen är artikel, starttid, mängd att producera och uppehållstid. De övriga cellerna ska redigeras automatiskt när en artikel väljs. Starttiden bör vara redigerbar

för att man använder sig av olika starttider beroende på vilken dag och vilken produkt som ska tillverkas. Under körordningssidan finns en knapp för att se produktionslistorna som genererats från körordningen, se figur 5.5. Både sidan för körordning och produktionslista innehåller knappar där det går att välja linje för att man enkelt ska få en överblick av den linje som redigeras för tillfället, skulle alla linjer visas på samma sida kan det bli mycket scrollande upp och ner på sidan.

Körordning Tue May 19 2020 13:47:47 GMT+0200 (Central European Summer Time) Info Orderhantering Körordning Tillverkningsorder Historik											
Körordning Linje 20 Linje 35 Linje 37 Linje 39 Linje 50 Linje 51 Linje 52 Linje 53 Linje 60 Linje 61 Linje 62 Linje 93 Linje 95											
Körordning: Linje 51											
Ändra datum	Måndag 2020-05-18										Skapa
Artikel	849 ▼	1104 ▼	1896 ▼	847 ▼	870 ▼	811 ▼	847 ▼	3186 ▼	870 ▼	Välj ▼	Välj ▼
Tillgänglig prod. tid	09,35										
Starttid	18,30	21,44	21,46	22,26	00,25	02,13	05,32	07,56	09,22		
Produkt											
Kart/Pall	40	40	36	40	40	40	40	36	40		
	80	80	80	72	72	61	72	72	72		
Enh/min	48	48	48	55	48	45	55	55	48		
Kart./Timme	420	420	420	425	425	386	425	425	425		
Mängd att producera	1355	15	72	632	551	1089	1020	396	1428		
Uppehållstid till nästa			0,30		0,30	0,30		0,30	0,30		
Produktionstid	3,14	0,02	0,10	1,29	1,18	2,49	2,24	0,56	3,22		
Återstående produktionstid	6,51	6,41	5,03	4,58	1,29	-0,57	-2,96	-4,52	-8,44		

Figur 5.4. Körordning från exempelwebb. (Författarnas egen bild)

Körordning Tue May 19 2020 13:48:46 GMT+0200 (Central European Summer Time) Info Orderhantering Körordning Tillverkningsorder Historik											
Körordning Linje 20 Linje 35 Linje 37 Linje 39 Linje 50 Linje 51 Linje 52 Linje 53 Linje 60 Linje 61 Linje 62 Linje 93 Linje 95											
Linje 51											
Ändra datum	Måndag 2020-05-18										
Artikel	Sort			Kartonger	Timmar	Start	Start pkt	Slut pkt	Pallar	Kart	Kart/tim
849		19,94	80	1355	3,14	18,30	21,45	00,59	33	35	420
1104		0,19	80	15	0,02	21,44	00,59	01,01	0	15	420
1896		0,90	80	72	0,10	21,46	01,01	01,11	2	0	420
847		8,78	72	632	1,29	22,26	01,41	03,10	15	32	425
870		7,65	72	551	1,18	24,25	03,40	04,58	13	31	425
811		17,85	61	1089	2,49	02,13	05,28	08,17	27	9	386
847		8,78	72	1020	2,24	05,32	08,47	11,11	25	20	425
3186		5,50	72	396	0,56	07,56	11,11	12,07	11	0	425
870		19,83	72	1428	3,22	09,22	12,37	15,59	35	28	425

Figur 5.5. Produktionslista från exempelwebb. (Författarnas egen bild)

5.2.5 Tillverkningsorder

Den fjärde knappen under huvudmenyn länkar till en sida för tillverkningsordrar, se figur 5.6. Under denna sida läggs alla tillverkningsordrar som skapats. Här finns knappar där man kan sortera ut om man vill se alla tillverkningsordrar, grupperade som 8703 eller tillverkningsordrar för varje tapp. Under de olika sorterade sidorna finns en knapp för att skapa tillverkningsordrar, ordena skapas med information från produktionslistan. Det finns även en knapp för att lägga in tillverkningsordrar manuellt. Det är exempelvis bra i det fall där distributionen behöver skapa tillverkningsordrar innan produktionslistan är färdig. Det finns knappar för att kunna hantera färdiga tillverkningsordrar ifall distributionen märker att det är något som behöver ändras på. Användaren behöver först markera en tillverkningsorder och sedan trycka på till exempel redigera för att kunna ändra något i en order. Till sist finns en knapp för att skicka de färdiga tillverkningsordrarna till produktion, tanken med den knappen är att allt ska kunna ändras och kontrolleras innan ordena skickas iväg. När man trycker på skicka TO kommer en popup-sida fram där användaren kan välja att godkänna och skicka enstaka ordrar eller hela orderlistan på en gång. Popup-sidan är skapad med en så kallad modal, se bilaga 2.

Tillverkningsorder											
Thu May 21 2020 21:40:17 GMT+0200 (centraleuropeisk sommartid)											
TO TO 8703 TO Tapp											
Generera TO Ny TO Redigera Ta bort Ändra datum Måndag 2020 Skicka TO till produktion											
Markera TO	Order ID	Tapp	Linje ▼	Status	Artikel	Önskat ant.	Producerat ant.	Beskrivning	Starttid	Starttid pkt.	Sluttid pkt.
☐	202004000011	0	35	Ej komplett		46.23	-		19.50	21.11	09.15
☐	202004000014	0	39	Ej komplett		35.86	-		19.30	21.05	03.43
☐	202004000015	0	39	Ej komplett		3.93	-		02.38	04.13	04.45
☒	202004000018	0	51	Ej komplett		18.03	-		18.30	21.45	00.59
☐	202004000019	0	51	Ej komplett		8.78	-		22.26	01.41	03.10

Figur 5.6. Tillverkningsordersida från exempelwebb. (Författarnas egen bild)

5.2.6 Historik och datum

Sist i menyn finns en sida för historik, där kan man se tillverkningsordrar som har producerats och är färdiga. De tillverkningsordrar som ska läggas in under historiken är de ordrar som har producerats under tidigare dagar än den nuvarande produktionsdagen.

Under varje sida förutom infosidan finns en knapp för att ändra datum. När användaren ändrar datum kommer alla sidors datum att ändras, det skulle lätt bli rörigt om alla olika sidor kanske skulle ha olika datum till slut på grund av att användaren har ändrat datum fram och tillbaka. Den sida som inte ska beröras av att datum ändras är sidan för historik, den sidan bör hantera datum för sig. Även för att ändra datum används en modal där användaren får en popup, där finns en kalender som man ställer in datumet på.

5.3 Kommunikation mellan webb och databas

För att kunna skriva mellan en webbsida och en SQL databas så används Javascript där kommunikationen programmeras och hanteras av plug-in program som tillför funktioner till Javascriptprogrammet. För att köra extra funktioner till Javascript används en plattform som heter node.js och genom denna installeras färdiga drivers som hanterar hämtning från databaser samt skrivning till webbsidor. De som använts i detta exempel är Express och MSSQL, se bilaga 3.

Med MSSQL drivern anropas kommunikationsvägen mellan webbsidan och databasen med användare, lösenord, server, databas och vilken port den läser genom. Sedan ges kommandon på vad som skall läsas eller skrivas till/från databasen med innehåll av vilka kolumner som skall ingå samt från vilken tabell datan skall hämtas ur. Denna data skickas sedan till den angivna webbsidan med hjälp av Express drivern.

6 Återkoppling

Här ges ett exempel på hur information från produktionen kan hanteras och presenteras.

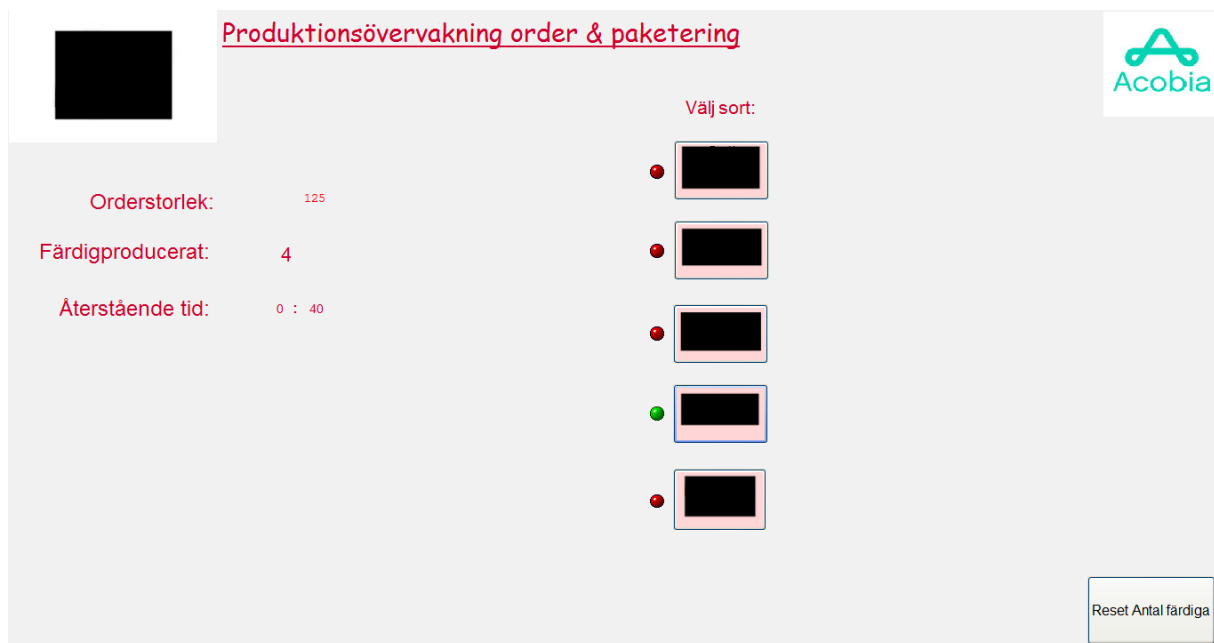
6.1 Simulerad återkoppling

På företaget används Mitsubishi PLC som är kopplade till ett övervakningssystem som heter CitectSCADA. PLC:erna sköter maskinfunktionerna och genom Citect kan man då alltså övervaka detta, se flödet och styra maskinerna så att allt fungerar. För distributionens del så är inte styrning av produktionen aktuell då de bara vill veta produktionsstatus för de olika orderarna. Vid packdelen av linorna, där produkten läggs i påse för att sedan läggas i en kartong finns det räknare som har koll på antalet färdiga enheter. Denna variabel kan användas för att visualisera vad som är färdigtillverkat, hur mycket som återstår av en order och jämföras mot orderns totala storlek för att med hjälp av de olika sorternas fasta produktionshastigheter ta fram återstående tid av en order.

6.2 Simuleringsexempel

Ett exempel har skapats i GX works 2 som är en programvara från Mitsubishi som används för att programmera deras PLC, se bilaga 4 som visar koden till PLC-programmet. Där skapas en puls av två timers, en TOF och en TON, denna pulserar med den frekvens som tillhör den valda produkten vilket ska motsvara räknaren. En puls motsvarar en kartong med produkter. Denna puls går sedan till en räknare som lägger resultatet på en global variabel (Varde).

På en skärm väljs en sort som skall köras, se figur 6.1, detta val görs bara i denna simulering då det i produktionen hos företaget hade fått värdena från tillverkningsordrar som ligger i en databas. När en order är vald läggs de tillhörande fasta värdena i respektive variabel som behandlar de olika beräkningarna. På skärmen visas då orderns storlek, antal färdiga kartonger och återstående tid.



Figur 6.1. Produktionsövervakning CitectSCADA. (Författarnas egen bild)

Variablerna i PLC:n läses av en OPC server som behandlar dessa och ger CitectSCADA tillgång till dem. Från Citect kan man både läsa och skriva till variablerna för att styra processen.

För produktionens skull i paketeringen ger denna skärm en överblick på vad som körs och hur en order ligger till.

För distributionens skull så är denna räknarvariabel användbar då de får en exakt siffra på hur många färdiga kartonger de har av en sort som de kan jämföra mot hur mycket de skickat iväg på lastbilarna och få fram det aktuella lagersaldot. Som de sedan läser av när de ska skapa nästkommande dags produktionslista.

Det är ganska komplicerat att få variabler från PLC in till en webbsida med liveuppdatering men exemplet som visas här visar de första stegen. En slutlig version skulle kunna visa all den nödvändiga informationen i en flik i den föreslagna webbsidan som sköter övrig orderhantering.

6.3 Hårdvara och inkoppling

I exemplet ovan användes en OPC-server men om ett liknande exempel skall införas på företaget sker kommunikationen mellan PLC och Citect direkt via ethernet utan en OPC-server mellan dem. Det har fungerat stabilt under de tester som har utförts och det är ett smidigt sätt då det är ett fåtal inställningar som behöver göras i Citect Studio för att få till en överföring av data mellan PLC:n och programmet, se bilaga 5. Mitsubishi Q12HCPU som användes under det här arbetet har inte en ethernet-port inbyggd, därför behövdes en extern modul kopplas ihop till PLC:n. Den ethernet-modul som användes heter Mitsubishi QJ71E71-100. För att kunna överföra data mellan Citect och PLC:n behövs även inställningar göras i GX Works2 som sedan överförs till PLC:n. Se bilaga 6 för hur dessa inställningar gjordes.

Det finns flera sätt att överföra ett skapat program till Q12HCPU. Det enklaste sättet är att använda sig av USB, här behövdes inga inställningar göras utan det är bara att koppla i kabeln och starta överföringen. Det är ett bra sätt om det snabbt ska överföras ett program. Ett annat sätt är att använda sig av ethernet. Det kräver att man gör fler inställningar för att det ska gå att få kontakt med CPU:n, se bilaga 7 för hur inställningarna är gjorda i Transfer Setup. Det finns CPU:er som har ethernet-portar inbyggda på CPU-modulen och det är möjligt att de inställningarna är något lättare att göra.

För att bara kommunicera mellan Citect och PLC:n via ethernet behövs egentligen inte inställningarna i Transfer Setup göras, det räcker att göra inställningarna som finns i bilaga 6. Anledningen till valet att även överföra program från GX Works2 till PLC:n via ethernet är för att undvika att ha en USB-kabel inkopplad. Detta gjordes dels för att vara helt säker på att det var ethernet som kommunikationen mellan Citect och PLC:n har skett genom och inte av misstag via USB, men även för att slippa att koppla i och ur en USB-sladd när ett program ska överföras.

Hela PLC-systemet består av fem delar, se tabell 6.1 för vilka märken och modeller som används. Kärnan i en PLC är CPU-modulen, det är den som programmeras och utför allt arbete. För kommunikationen står en ethernet-modul. Alla moduler behöver kopplas ihop på något sätt och det görs via en basenhet. För strömförsörjning till PLC:n används två delar. En strömförsörjning som omvandlar 220 volt från vägguttaget till 24 volt. Det finns även en strömförsörjningsmodul som sedan omvandlar 24 volt till 5 volt vilket krävs av Q-seriens olika moduler som kopplas i basenheten. Det finns olika spänningsuttag på strömförsörjningsmodulen för delar som kräver 24 volt istället. För en bild på PLC-systemet som använts se bilaga 8.

Tabell 6.1: PLC hårdvara

Hårdvara:	Märke och modell
CPU	Mitsubishi Q12HCPU
Ethernetmodul	Mitsubishi QJ71E71-100
Strömförsörjningsmodul	Mitsubishi Q63P
Strömförsörjning	Traco Power TSL 060-124
Basenhet	Mitsubishi Q38DB

7 Resultat

Utifrån intervjuer och undersökning av de befintliga systemen har ett slutligt resultat tagits fram som är applicerbart med de system som finns tillgängliga och används idag hos företaget.

Webbsidan som beskrivits i tidigare avsnitt hade gett ett enklare och säkrare arbetssätt då all hantering av inkommande ordrar sker i samma program. När alla inputs som krävs för att skapa körordningen är samlade på ett gemensamt ställe blir det manuella steget för att skapa denna körordning smidigare och risken för fel minskar.

Körordningen skapas i en sida i webben där artikelnummer och mängden som skall tillverkas väljs in, den tillhörande fasta datan till en artikel läses in och ger en produktionstid. Detta görs med alla sorter som skall köras och detta blir slutligen körordningen. När denna del är klar skapas en produktionslista med ordningen baserad på körordningen. Det sista steget från när produktionslistan översätts till tillverkningsordrar sker automatiskt. Det innebär att produktionslistan komprimeras och endast visar den information som är nödvändig i produktionen. Om tre ordrar i produktionslistan är av samma sort men olika artikelnummer kan dessa med fördel sammansättas till en gemensam tillverkningsorder som visar den totala storleken på ordern och det totala antalet satser som krävs för att tillverka dessa tre ordrar.

När tillverkningsordrarna är klara kan dessa kontrolleras en sista gång innan de godkänns. Vid godkännande så skickas dessa till företaget produktionsdatabas varifrån de sedan läses in till den befintliga webbsidan som finns i tapp 5 i produktionen, denna webbsida behöver införas i alla tappar för att systemet skall fungera till alla produktionslinjerna.

Det tidigare arbetssättet som bestått av att ringa från distributionen till produktionen för att lägga till eller ta bort tillverkningsordrar kommer nu ske online via webbsidan då redigering kan göras under produktionens gång.

Under produktionens gång så använder många olika delar i produktionen sig av produktionslistorna för att hålla koll och se vad som körs och vad som kommer efteråt. De kommer att få tillgång till listorna genom webben och på så sätt kommer alla berörda avdelningar att se och uppmärksamma uppdatering av körordningen snabbare.

Alla produktionslistor och tillverkningsordrar som skapats kommer att lagras i en databas för att föra historik och ge möjlighet till att gå tillbaka och se vad som körts tidigare. Detta kan användas om något fel uppstår med inkommande beställningar. Då kan man se föregående veckas eller liknande tidpunkts ordrar och produktionslistor för att kunna göra en uppskattning av vad som bör produceras.

Webbsidan behöver ha olika auktoritetsnivåer för att bara behörig personal skall kunna hantera delarna med planering och godkännanden eller för dem som använder sig av de olika listorna i produktionen. Detta kommer att hanteras med olika login som ger den berörda personalen tillgång till de som de har rättighet till.

Återkopplingen från produktionen som visar vad som körs och ger möjlighet till att räkna ut lagersaldon skall presenteras under en flik i webbsidan där planerarna kan kolla när de skapar körordningen. Dessa variabler kan även användas för att visa vilken produkt och order som körs i varje linje och hur lång tid det återstår av en order. Exempelvis på en skärm vid paketeringsdelen då de ger kontroll och överblick på när det är dags att byta sort.

8 Diskussion

Arbetet med att förbättra företagets interna ordersystem visade sig vara en väldigt stor och avancerad uppgift. Det började i liten skala då endast den manuella hanteringen med papperslappar mellan avdelningarna skulle bort men växte snabbt till ett helt system som skulle plocka in beställningar och lagersaldon till en gemensam webb. I denna webb som skall ersätta dagens Excel strävades det efter att generera produktionslistorna automatiskt och få bort alla manuella steg förutom ett godkännande innan det skickades ut till produktionen. Denna automatisering visade sig vara ett komplicerat steg då alla olika beställningar kommer in till företaget genom flera olika system som deras egna Vudiax system, mejl och räkning i lagret. Även skapandet av körordning är svårt att göra automatiskt eftersom det bygger på manuellt arbete och erfarenhet. Istället för att sträva efter ett helt automatiskt system som hade krävt många stora förändringar så hölls projektet inom ramarna för vad som är möjligt med de befintliga systemen men ändå skapar en modernare och smidigare hantering av inkommande beställningar till skapandet av färdiga tillverkningsordrar.

För att nå alla delar i kravspecifikationen krävs ett större arbete med specialiteter inom flera olika områden som slutligen kan resultera i ett nytt ordersystem som sköter alla berörda delar. Vudiax som idag används av de svenska säljarna för att lägga beställningar skulle också behöva ingå i det nya systemet tillsammans med storhushållskunder och exportbeställningar. När alla beställningar då samlas på ett ställe kan körordningen bestämmas med hjälp av ett program som läser av alla beställningars storlekar och avgångstider och genererar en lämplig körordning. Genom att all data från beställningarna finns i databaser och hanteras genom Javascript skulle det vara möjligt att ta fram en kod i Javascript som sköter uppdelningen.

Utgångspunkten för denna analys var att ta fram ett modernare och mer användarvänligt orderhanteringssystem, det medför inte bara ett enklare och bättre system utan de underlättar för personalen på flera sätt. När uppladdning och redigering av tillverkningsordrar kan ske online istället för telefonsamtal och pappershantering ger det mer tid till övriga arbetsuppgifter och har positiv inverkan på miljön samt medför att stress minskar. Med ett säkrare och enklare arbetssätt som medför färre misstag och minskar felproduktionen gynnas både miljön och ekonomin positivt. Det blir ett bättre flöde från inkommande ordrar till utgående leveranser som ger hela företaget ett positivt resultat.

9 Vidareutveckling

Det första steget för att gå mot ett helt automatiserat flöde är att skapa ett system som behandlar alla ordrar från Vudiax, export och STH tillsammans med lagersaldon i samma program. Ett nytt system skulle kunna behandla beställningar genom en onlinetjänst som skickar data till en databas på företaget som sedan läses in och presenteras i webbprogrammet som hanterar beställningarna. I onlinesystemet skulle kunderna kunna lägga in beställningar själva och om onlinetjänsten är kopplad till företagets orderhanteringssystem så kommer den datan in via onlinetjänsten istället för via mejl. Beställningarna från Vudiax och den aktuella lagerstatusen läses in tillsammans med de externa beställningarna och skapar en stor gemensam behovslista som visar alla ordrar som kommer in på ett samlat sätt.

Genereringen av körordning automatiskt skulle kunna vara möjlig i framtiden om denna gemensamma webb som hanterar alla ordrar skapas. Då en order har sort, mängd och avgångstid bundet till sig så kan dessa variabler användas i ett program som sköter uppdelningen till körordning automatiskt.

Eftersom all info som orderlistor, produktionslistor och tillverkningsordrar finns elektroniskt skulle de som använder sig av dem i produktionen ha tillgång till dessa genom en APP eller webbsida och på så sätt få bort all pappershantering och genom detta få bättre och snabbare koll på om ändringar görs.

10 Slutsats

Lösningförslaget som föreslås i denna rapport är anpassat utifrån att behålla de system som används idag för beställningar och behålla den manuella uppläggnings av körordningar. Det som föreslås att förnyas är hur denna hantering ska ske. Med den föreslagna webbsidan som ersätter de gamla Excel-dokumentet kommer det bli lättare att överblicka, skapa och redigera ordrar för att sedan automatiskt tanka ut dem som tillverkningsordrar till produktionslinjernas tappar.

Storleken och nivån på att skapa det föreslagna systemet känns rimligt och det kommer gynna företaget när det appliceras.

11 Källförteckning

- [1] P. Wilton, J. W. Colby, Beginning SQL. 1st ed., New Jersey, USA: John Wiley & Sons Inc. 2005. [Online] Tillgänglig: <https://ebookcentral.proquest.com/lib/chalmers/detail.action?docID=226434&query=beginning+sql>, hämtad: 2020-04-28.
- [2] Refsnes Data, "HTML Introduction," 2020. [Online]. Tillgänglig: https://www.w3schools.com/html/html_intro.asp, hämtad: 2020-05-12.
- [3] C. Coremans, HTML: A Beginner's Tutorial. 1st ed., Brainy Software Inc. 2015. Tillgänglig: <https://ebookcentral.proquest.com/lib/chalmers/reader.action?docID=4339838>, hämtad: 2020-06-05.
- [4] Refsnes Data, "CSS Introduction," 2020. [Online]. Tillgänglig: https://www.w3schools.com/css/css_intro.asp, hämtad: 2020-05-12.
- [5] J. McPeak, P. Wilton, Beginning Javascript. 5th ed., New Jersey, USA: John Wiley & Sons Inc. 2015. [Online] Tillgänglig: <https://ebookcentral.proquest.com/lib/chalmers/reader.action?docID=479865>, hämtad: 2020-04-28.
- [6] R. Fergusson, Beginning Javascript. 3rd ed., New York, USA: Apress. 2019. [Online] Tillgänglig: <https://link-springer-com.proxy.lib.chalmers.se/content/pdf/10.1007%2F978-1-4842-4395-4.pdf>, hämtad: 2020-04-28.
- [7] Schneider Electric, "CitectSCADA," 2020. [Online]. Tillgänglig: <https://www.se.com/se/sv/product-range-presentation/60288-citectscada/>, hämtad: 2020-05-12.
- [8] D. H. Hanssen, Programmable Logic Controllers. 1st ed., New Jersey, USA: John Wiley & Sons Inc. 2015. [Online] Tillgänglig: <https://ebookcentral.proquest.com/lib/chalmers/reader.action?docID=4040112&query=programmable+logic+controller>, hämtad: 2020-04-28.
- [9] Livsmedelsverket, "Frivillig information," 2019. [Online]. Tillgänglig: <https://kontrollwiki.livsmedelsverket.se/artikel/52/frivillig-information>, hämtad: 2020-04-27.

Bilagor

Bilaga 1, Kod för sidebar

```
<link rel="stylesheet" type="text/css" href="sidebar1.css">

<div class="sidenav">
  <a href="Index.html">Info</a>
  <a href="Orderhantering.html">Orderhantering</a>
  <a href="Korordning.html">Körordning</a>
  <a href="Tillverkningsorder.html">Tillverkningsorder</a>
  <a href="Historik.html">Historik</a>
</div>
```

```
.sidenav {
  border-right: 2px solid #000080;
  border-radius: 2px;
  height: 90%;
  width: 170px;
  position: fixed;
  margin: 85px 0px;
  background-color: #000080;
}

.sidenav a {
  background-color: #000080;
  padding: 6px 6px 4px 14px;
  text-decoration: none;
  font-size: 20px;
  color: #000080;
  display: block;
}

.sidenav a:hover {
  background-color: #000080;
  color: #000080;
}
```

Bilaga 2, Kod för modal

```
<div id="SkickaModal" class="modal">
  <div class="modal-innehall">
    <span class="close">&times;</span>
    
  </div>
</div>

<script>
var modal = document.getElementById("SkickaModal");

var button = document.getElementById("buttonSkicka");

var stang = document.getElementsByClassName("close")[0];

button.onclick = function() {
  modal.style.display = "block";
}

stang.onclick = function() {
  modal.style.display = "none";
}
</script>
```

```
.modal {
  display: none;
  position: fixed;
  padding-top: 80px;
  left: 0;
  top: 0;
  width: 100%;
  height: 100%;
  background-color: rgba(0,0,0,0.5);
}

.modal-innehall {
  background-color: #ffffff;
  margin: auto;
  padding: 18px;
  border: 1px solid rgb(130, 130, 130);
  width: 1350px;
}

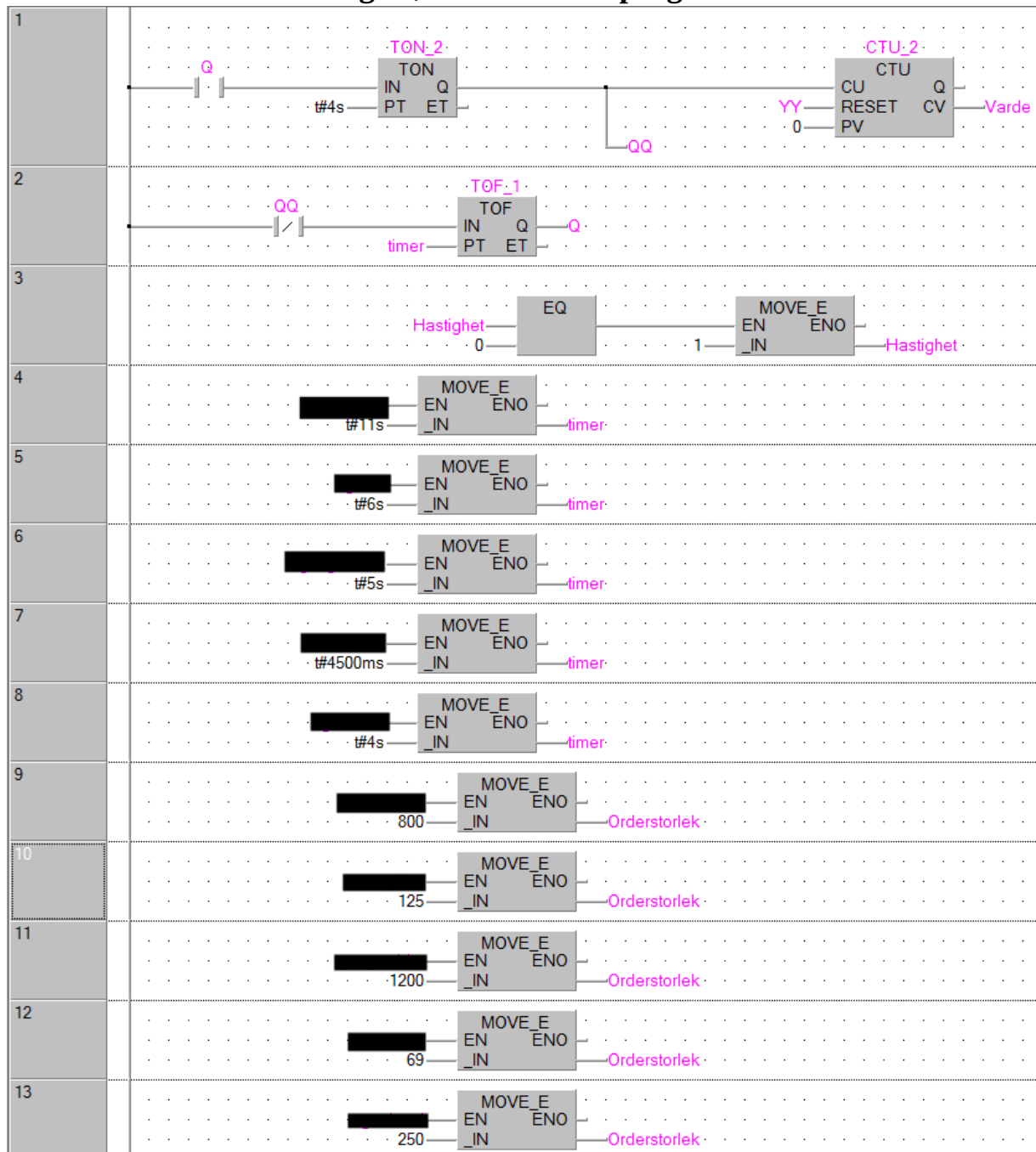
.close {
  color: rgb(130, 130, 130);
  float: right;
  font-weight: bold;
  font-size: 24px;
}

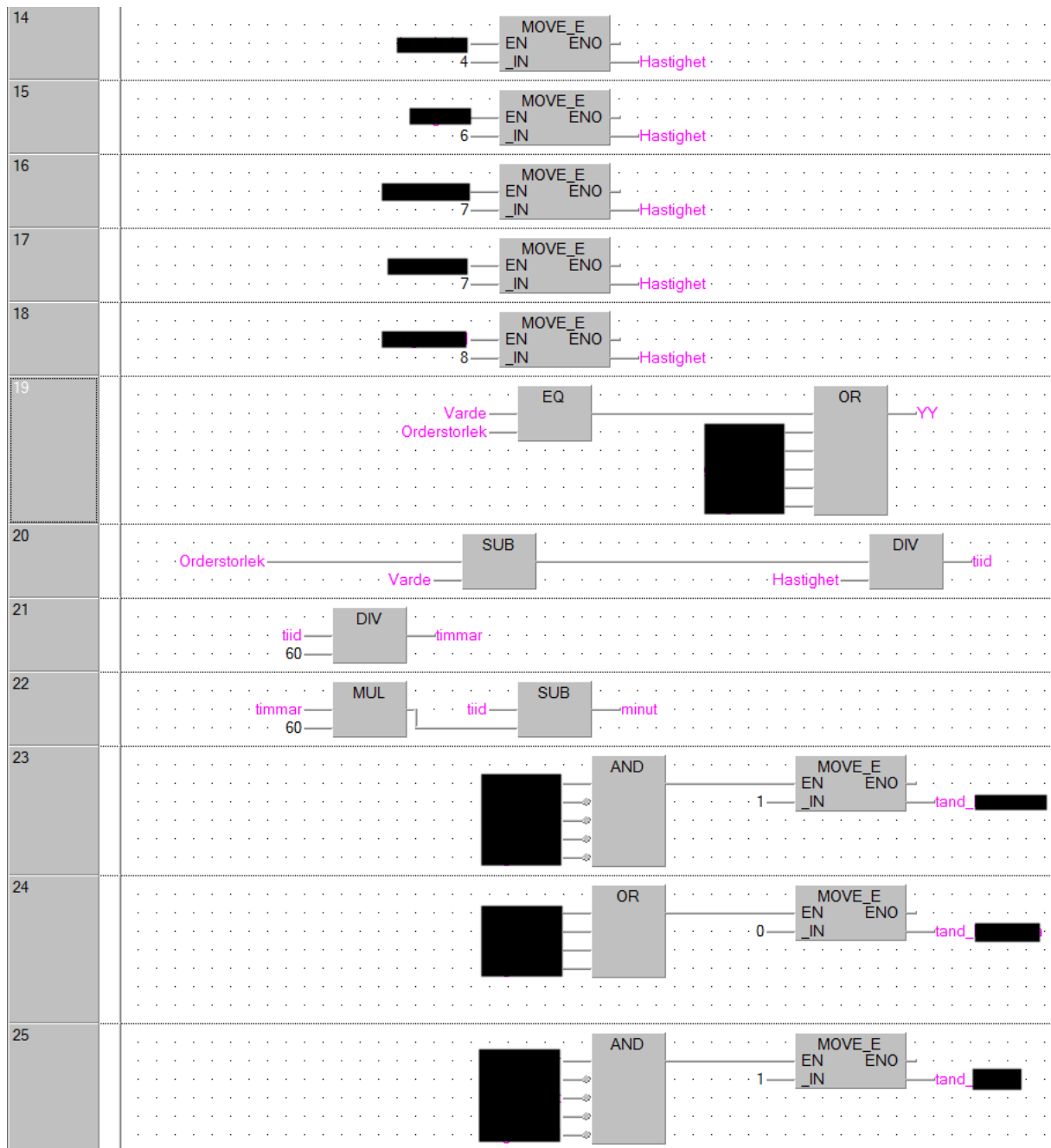
.close:hover{
  color: rgb(20, 20, 20);
  cursor: pointer;
}
```

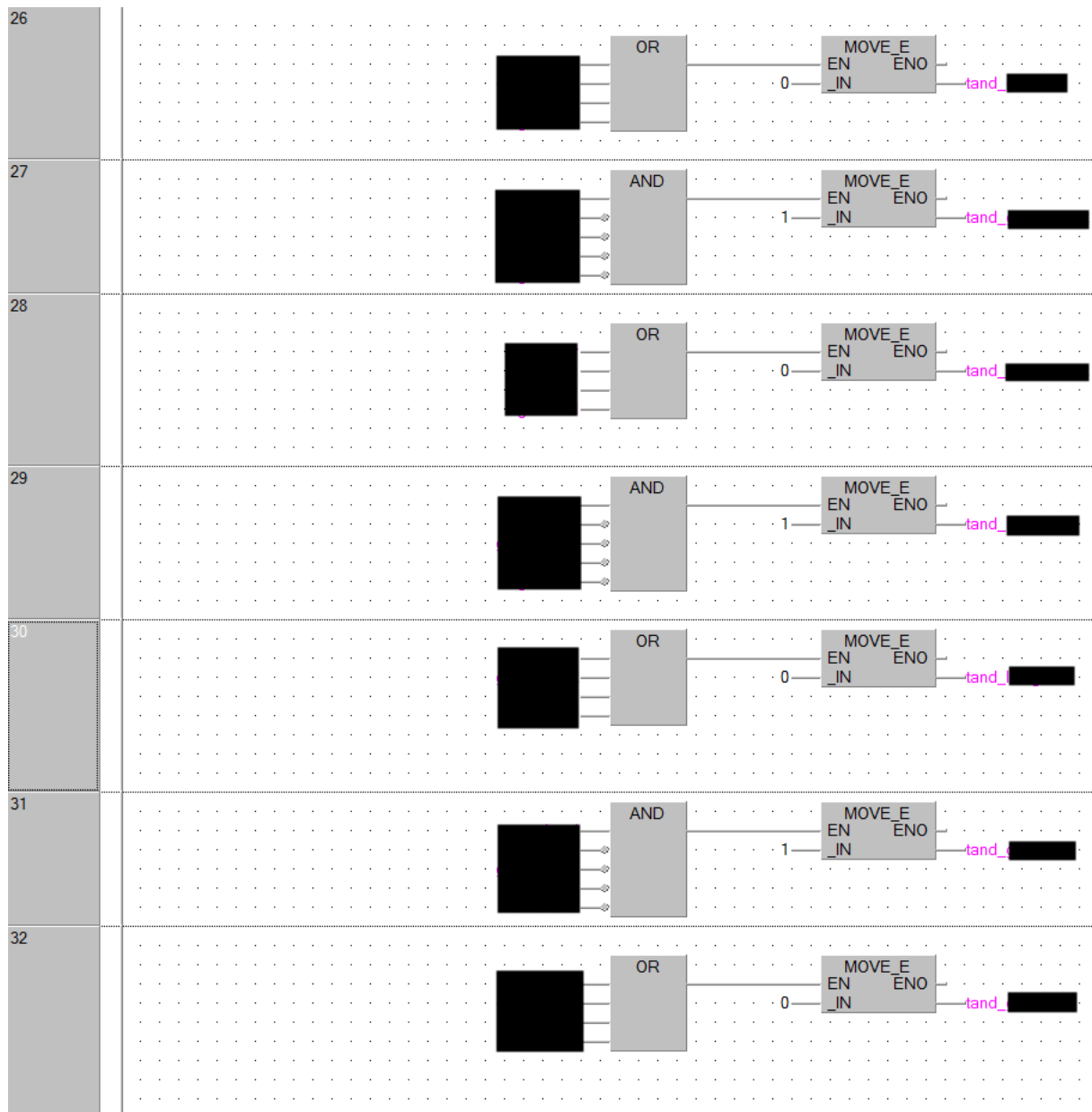

Bilaga 3, Kod för kommunikation mellan webb och databas

```
1 const express = require('express');
2 const app = express();
3 const bodyParser = require("body-parser");
4 const path = require('path');
5
6 app.use(bodyParser.urlencoded({ extended: false}));
7
8
9 app.get('/', function (req, res){
10   res.sendFile(path.join(__dirname + '/sortera.html'));
11 });
12
13 app.post('/connect_', function (req, res) {
14
15   var sql = require("mysql");
16
17   const config = {
18     user: 'Exjobb',
19     password: 'Ludwig.ojalai',
20     server: 'localhost',
21     database: 'Vudiax',
22     port: 1433
23   };
24
25   sql.connect(config, function(err){
26     if (err) console.log(err);
27
28     let sqlRequest = new sql.Request();
29     let Linjex = req.body.Linjen;
30     let [ ] = req.body.I [ ];
31     let sqlSelectQuery = '';
32     let sqlInsertQuery = '';
33
34     sqlSelectQuery = 'Select Orderid, Artikelnr, Artikelnamn, Kartonger, Godsadressnr, Godsadress, Avgangstid, Linje, Datum';
35     sqlInsertQuery = 'insert INTO vudiax_ut_tapp_5 (Orderid, Artikelnr, Artikelnamn, Kartonger, Godsadressnr, Godsadress, Avgangstid, Linje, Datum)'+ sqlSelectQuery;
36
37
38     sqlInsertQuery+= ' From vudiax_in WHERE 1=1 '
39     sqlSelectQuery+= ' From vudiax_in WHERE 1=1 '
40
41     if (Linjex != 0 ){
42       sqlInsertQuery+= ' AND Linje = ' + parseInt(Linjex);
43       sqlSelectQuery+= ' AND Linje = ' + parseInt(Linjex);
44     }
45     if ( [ ] != 0){
46       sqlInsertQuery += " AND Artikelnamn = '" + [ ] + "'";
47       sqlSelectQuery += " AND Artikelnamn = '" + [ ] + "'";
48     }
49
50     console.log(sqlInsertQuery);
51
52     sqlRequest.query(sqlSelectQuery, function(err, data){
53
54       if (err) console.log(err)
55       console.table(data.recordset);
56       console.log(JSON.stringify(data));
57       console.table(data.recordset);
58       console.log(data.rowsAffected);
59       console.log(data.recordset[0]);
60
61       let h="<h1 style='font-size:3em;background-color: rgb(191, 49, 27);margin:20px;padding:20px;text-align: left;color: rgb(243, 210, 61);'><img src='\" + [ ] + ".bild.png\"
62 let str= '<table style='\"width:100%;border:1px solid rgb(191, 49, 27);'>' + '<tr>'
63   + '<th style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + 'Orderid' + '</th>'
64   + '<th style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + 'Artikelnr' + '</th>'
65   + '<th style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + 'Artikelnamn' + '</th>'
66   + '<th style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + 'Kartonger' + '</th>'
67   + '<th style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + 'Godsadressnr' + '</th>'
68   + '<th style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + 'Godsadress' + '</th>'
69   + '<th style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + 'Avgangstid' + '</th>'
70   + '<th style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + 'Linje' + '</th>'
71   + '<th style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + 'Datum' + '</th>'
72   + '</tr>';
73
74 let row = '';
75
76 for (let j=0;j<data.recordset.length;j++){
77   row=row + '<tr>'
78   + '<td style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + data.recordset[j].Orderid + '</td>'
79   + '<td style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + data.recordset[j].Artikelnr + '</td>'
80   + '<td style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + data.recordset[j].Artikelnamn + '</td>'
81   + '<td style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + data.recordset[j].Kartonger + '</td>'
82   + '<td style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + data.recordset[j].Godsadressnr + '</td>'
83   + '<td style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + data.recordset[j].Godsadress + '</td>'
84   + '<td style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + data.recordset[j].Avgangstid + '</td>'
85   + '<td style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + data.recordset[j].Linje + '</td>'
86   + '<td style='\"width:50px;border:1px solid rgb(191, 49, 27);'>' + data.recordset[j].Datum + '</td>'
87   + '</tr>';
88
89 }
90
91 let knapp = '<form>' + '<input type="button" id="Exportera" value="Exportera" />' + '</form>';
92 str=str + row + '</table>';
93 res.send(h+str+knapp);
94
95 /* sqlRequest.Exportera(sqlInsertQuery);*/
96 sql.close();
97
98 });
99
100
101
102 });
103 });
```

Bilaga 4, Kod för PLC-program







Bilaga 5, Inställningar för kommunikation mellan CitectSCADA och PLC

Citect Studio - PLC_Ethernet [Active Project]

Topology Computers Edit Profiles I/O Devices Components & Mapping

Save Discard Copy Paste Delete Row(s) Export All Import All Export Tags Import Tags Refresh Tags New Device Driver Reference Help

Row	Server Name	Name	Number	Address	Protocol	Port Name	Startup Mode	Priority	Memory	Read-Only	Comment	Project
1	IOServer1	Internal	1		MELSECQ		Primary	1	TRUE		Internal memory	PLC_Ethernet
2	IOServer1	Cicode	2		CICODE		Primary	1	FALSE		I/O Device for cal	PLC_Ethernet
3	IOServer1	IODEV	3	255	MELSECQ	PORT1_BOARD1	Primary	1	FALSE	FALSE		PLC_Ethernet

Citect Studio - PLC_Ethernet [Active Project]

Topology Computers Edit Profiles I/O Devices Components & Mapping

Save Discard Copy Paste Delete Row(s) Export All Import All

Boards

Row	Server Name	Board Name	Board Type	Address	I/O Port	Interrupt	Special Opt	Comment	Project
1	IOServer1	BOARD1	TCPIP	0					PLC_Ethernet

Bilaga 6, Inställningar för kommunikation via ethernet för PLC

Network Parameter

	Module 1	Module 2	Module 3	Module 4
Network Type	Ethernet	None	None	None
Start I/O No.	0000			
Network No.	1			
Total Stations				
Group No.	1			
Station No.	1			
Mode	Online			
	Operation Setting			
	Initial Setting			
	Open Setting			
	Router Relay Parameter			
	Station No. <-> IP Information			
	FTP Parameters			
	E-mail Setting			
	Interrupt Settings			

Necessary Setting(No Setting / Already Set) Set if it is needed(No Setting / Already Set)

Interlink Transmission Parameters Start I/O No. : Valid Module During Other Station Access 1

Please input 16-point unit(HEX) to start I/O No. in which module is mounted.

Acknowledge XY Assignment **Routing Parameters** Assignment Image Group Setting... Check End Cancel

Print Window... Print Window Preview

	Protocol	Open System	Fixed Buffer	Fixed Buffer Communication	Pairing Open	Existence Confirmation	Host Station Port No.
1	TCP	Unpassive	Send	Procedure Exist	Disable	No Confirm	1280
2							
3							
4							

Ethernet Operation Setting

Communication Data Code
☒ Binary Code
☐ ASCII Code

Initial Timing
☐ Do not wait for OPEN (Communications impossible at STOP time)
☒ Always wait for OPEN (Communication possible at STOP time)

IP Address Setting
 Input Format DEC
 IP Address 192 168 3 50

Send Frame Setting
☒ Ethernet(V2.0)
☐ IEEE802.3

☐ Enable Online Change

TCP Existence Confirmation Setting
☐ Use the KeepAlive
☒ Use the Ping

End Cancel

PLC Parameter

Q Parameter Setting

PLC Name | PLC System | PLC File | PLC RAS | Boot File | Program | SFC | Device | I/O Assignment | Multiple CPU Setting

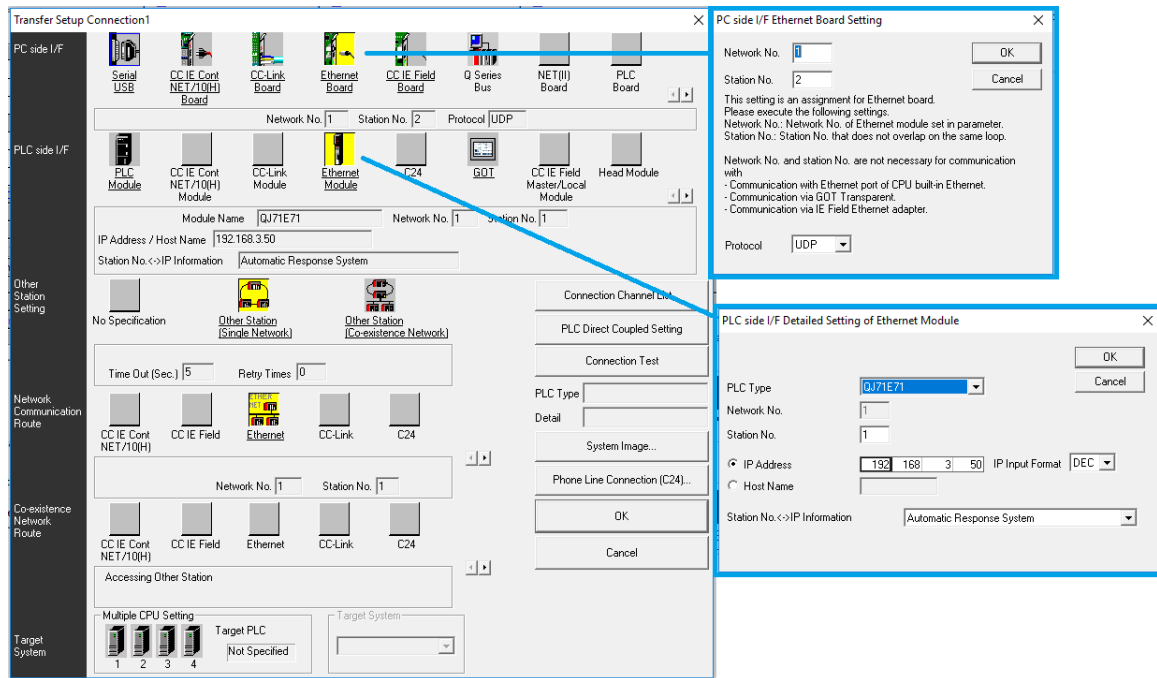
I/O Assignment(*1)

No.	Slot	Type	Model Name	Points	Start XY	Switch Setting
0	PLC	PLC	Q12H-CPU			
1	0(0-0)	Intelligent	QJ71E71	32Points	0000	
2	1(0-1)					
3	2(0-2)					
4	3(0-3)					
5	4(0-4)					
6	5(0-5)					
7	6(0-6)					

Assigning the I/O address is not necessary as the CPU does it automatically.
 Leaving this setting blank will not cause an error to occur.

Detailed Setting
 Select PLC type
 New Module

Bilaga 7, Inställningar för programöverföring till PLC



Bilaga 8, PLC-hårdvara

