2021

The eggficient Prinzen

Insights in the throughput time.

UNIVERISITY OF TWENTE
BACHELOR THESIS INDUSTRIAL ENGINEERING AND MANAGEMENT (IEM)
FACULTY OF BEHAVIOURAL, MANAGEMENT AND SOCIAL SCIENCES (BMS)

BY JAAP LEUVERINK

PRINZEN | WEVERIJ 18, 7122 MS
Supervisor: Gayane Sedrakyan
2nd supervisor: Asad Abdiesfandani
Company supervisor: Mike Kiens

“Who controls the past controls the future: who controls the present controls the past”

~ George Orwell

Preface

The report you are about to read is the result of the bachelor thesis conducted at Prinzen to complete the bachelor Industrial Engineering and Management at the University of Twente. The thesis was under the supervision of the team leader operations and the affairs & resource officer of Prinzen.

I would like to thank my supervisors and all other employees, whether they were involved in my project or not, for allowing me to do my bachelor thesis at Prinzen. A special thanks to all the employees who contributed to the research and showing support. I felt, and still feel very welcome at Prinzen. Therefore, I am glad that our collaboration does not end with the thesis because of the opportunity to work for Prinzen.

Also, a special thanks go to the supervisor from the University of Twente, Gayane Sedrakyan. The thesis would not be the same without your fast reactions to my emails, the feedback, and the suggestions. I would also like to thank the second reader of this report, Asad Abdiesfandani.

At last, I Would like to thank my family and friends for always believing in me during my study and research, and allowing me the opportunity to make sure this thesis came to a good end. A special thanks goes to my buddies: Yorick Beekman, Siemen Kaak, and Jimmy van Santen. Your feedback and support improved my thesis and helped me to stay on track.

Jaap Leuverink

July 2021

Management summary

Introduction

Prinzen is a company based in Aalten, the Netherlands, and specialized in egg handling equipment. Prinzen is part of the Vencomatic Group, the group provides solutions in housing equipment, egg handling, and climate control. The Vencomatic group tries to deliver solutions with the lowest possible environmental impact while achieving the highest animal welfare levels. The specialty of Prinzen is a machine that produces a 99.7% points down setting of the eggs.

Prinzen wants to be eggficient, part of this plan is that they deliver machines to clients with a throughput time of 4, 8, or 12 weeks. However, the current situation is that they rarely meet those target values. This leads to the following action problem:

At this moment the throughput time is difficult to measure and inconsistent due to several reasons mentioned below, that is why Prinzen is not able to reach their target times of 4, 8, or 12 weeks.

Interviews with the stakeholders identified several possible problems for a too long throughput time, but all problems mentioned were based on feelings and observations from the practice. The current throughput time was not known, and this is necessary to know where the problem is occurring. Therefore the following main research question is answered in this thesis:

How can Prinzen get a clear image of what the current throughput time is concerning the planned throughput figures?

Solution method

To solve the core problem and answer the main research question a solution method is designed. The solution method is based on a summary of methods in theories found from the current academic literature combined with inputs from the practice. The method is easy to use, fits the characteristics of the case, is measurable, and has an added value for future use. It consists of 5 steps. The first step is to analyse the order process flow. The second step is to identify key performance indicators (KPIs). Since there is no data available for this particular company case the third step is to create dummy data. The fourth step is to create a dashboard with the output of all the previous steps. For this company case, the dashboard shows what could be monitored if the data was available. The dashboard also has a simulation function, which was a suggestion by Prinzen. The fifth and last step is to evaluate the output of the solution method with the stakeholders. The solution method is a mixed research method. It uses quantitative research for the analysis of the dummy data, but it also uses qualitative research for the analysis of the order process. The analysis of the order process is achieved by the use of interviews with Prinzen employees.

Implementation

The implementation is divided into three parts, KPIs, dummy data, and the dashboard. The KPIs were found in the literature and mentioned by stakeholders of the different departments in Prinzen. The KPIs are prioritized according to the analytical hierarchy process (AHP). This method is based on the pairwise comparison, the comparison is done by the problem owner. The following list, from highest to lowest prioritization, is the results of the AHP.

- Value-added time
- Average time per phase

- Number of orders in progress per phase
- Warning for too long throughput times
- Planned orders in progress
- Planned operation time

The second part of the implementation is the creation of the dummy data. The input for the dummy data came from interviews with the stakeholders of different departments in Prinzen. With the use of Visual Basic for Applications (VBA) the input data generated two sheets, one event log and two a capacity sheet. The event log is filled with orders and the times the departments are finished. The capacity sheet is filled with the number of full-time employees available each week per department in Prinzen.

The third part of the implementation is the generation of the KPIs for the dashboard. Each KPI is generated on its sheet, with the use of a VBA code. The table in the sheet is transformed into a graph and shown on the dashboard. All KPIs are implemented, except for the value-added time. For this KPI, not enough data was available at this moment. An extra feature is added to the dashboard which makes it possible for the user to simulate one day further in time. If the input data represents the current situation, then this feature can be used to see what happens if the team leader operations change certain parts of the order process.

Conclusion & recommendations

The research concludes that with the necessary data available the dashboard with its KPIs is able to give insight into the throughput time. It is unfortunate that the data is not available and that the dashboard is therefore based on dummy data only. Two recommendations were proposed to make sure that Prinzen is able to make a dashboard based on the real data of the order process. The first is to keep track of the order process with the use of an event log. The second recommendation is to label orders with their target time as soon as they are entered into the system. This label will be useful for the analysis of the throughput time.

Evaluation

This research has its limitations. The biggest limitation of the research is the lack of useful data. The data gathered via interviews is based on feelings and observations from the practice. While for this research it would be more useful to have already recorded data of the order process from the past. The observations could be erroneous, and observations change from day to day. The second limitation is the correctness of the data, this is a known problem for Prinzen. The data from the Enterprise Resource Planning (ERP) system shows something else than what happens in reality. A positive thing is that the dashboard has a sufficient capability to allow for issues detection earlier during the process.

Table of contents

List of figures

List of tables

List of Abbreviations

KPI(s)	= Key Performance Indicator(s)
AHP	= Analytical Hierarchy Process
VBA	= Visual Basic for Applications
ERP	= Enterprise Resource Planning
MTO	= Make To Order
SMART	= Specific, Measurable, Attainable, Realistic and Time-sensitive
FTE	= Full-Time Equivalent

1. Introduction

1.1.Company introduction

Prinzen is a company based in Aalten. Prinzen is a part of the Vencomatic Group. The Vencomatic Group consists of several companies. The Vencomatic Group provides solutions in housing equipment, egg handling, and climate control. They do this for any type of poultry. Their slogan is: “Think ahead with poultry people”. They try to deliver solutions with the lowest possible environmental impact while achieving the highest animal welfare levels.

Prinzen is the company that makes the egg handling equipment. Prinzen was founded in 1984, they soon started to develop their hatching egg packers and simple transfer. They developed a machine that produces 99.7% points down setting of the eggs, this is 2,7% higher than when it is manually done. When an egg is pointed down onto a tray the air pocket in the egg helps to keep the yolk centered, this reduces the risk of the egg spoiling. In hatching, this is very important, because more eggs hatching means more money for the farmer. The machines for the hatchery have a capacity of 12.000 to 30.000 eggs per hour. The consumption egg machines have a capacity of 18.000 to 80.000 eggs per hour. More information about the machines is found in appendix A.

Prinzen wants to promise their clients that they can deliver different machines in 4, 8, or 12 weeks. The current situation is that they rarely meet those target values. The throughput-time of the company is too long. The thesis aims to reduce the throughput time. This can be done by a key performance indicator that shows what the throughput time is for the whole company, a KPI about the throughput time per department, and a KPI that will show what the capacity is based on the throughput time.

1.2. Problem identification

A lot can be improved during all phases of the process at Prinzen according to the team leader production. This section presents the action problem. The core problem that needs to be solved is also identified. The core problem is determined via a problem cluster.

1.2.1. Action problem

The problem that arises at Prinzen is that the targets in relation to delivery times are not always met. The delivery times are mostly affected by the throughput times. The throughput time is defined as the time an order is in the company, that is from order entry till the order leaves the company. The throughput times that Prinzen uses as classifications are 4, 8, or 12 weeks, depending on the complexity of the order. The problem has been around for some time, and no one can say exactly where the problem originates. Prinzen wants to achieve throughput times of 4, 8, or 12 weeks because it wants to differentiate itself from the competition.

Conformance of the desired and actual processes

The industry asks for a short delivery time. The delivery time is mostly influenced by the throughput time. To become an industry leader, the plant manager of Prinzen set a norm for the throughput time. The norm that Prinzen wants to achieve is 4 weeks for standard machines, 8 weeks for multiple machines (combination of 2 or 3 machines), and 12 weeks for complete lines. What the reality exactly is, is difficult to say. The throughput time varies widely. Also, it is not clear where in the organization the problem occurs. The problem owner is the Team leader operations.

Action problem

At this moment the throughput time is difficult to measure and inconsistent due to several reasons mentioned below, that is why Prinzen is not able to reach their target times of 4, 8, or 12 weeks.

1.2.2. Problem cluster

To identify the core problem, all the causes of the action problem are analyzed and identified. The relationships to the root causes are further analyzed. Then they are shown in the problem cluster figure below the problems that are mentioned by the stakeholders. To identify the root causes conversations took place with representatives from various levels within the company. A description of the conversation can be found in Appendix B.

- **The non-flexibility of resource planning in production.**
  Prinzen made a competence matrix, in this matrix every production worker is assessed on his skills on a certain machine or task. Then according to that matrix, production workers are planned on building a machine with a certain productivity percentage. Someone with a lot of experience will be planned with 100% productivity, at the same time a new person will only be planned with a productivity of 50%. There are too few people who have a productivity of 100% on diverse machines.

- **Production workers are not technical enough.**
  The production workers of Prinzen learned how to build a certain machine, however, this skill is not sound i.e. based on solid experience and technical knowledge. They learned the tricks on how to build a machine. There is a lack of technicality.

- **The manipulation of bills of materials causes problems.**
  At Prinzen the machines are standardized, but they can be changed according to the wishes of the customers. The employees of work preparation change the bills of materials in such a way that the wishes of the customers are considered. Those changes in the bills of materials cause problems in the production, due to missing, wrong or too many parts.

- **Suppliers do not deliver on time.**
  A lot of suppliers of Prinzen do not deliver on time. The materials arrive too early or too late.

- **Suppliers deliver poor quality.**
  The materials delivered by the suppliers are damaged or not according to the technical standard. Therefore Prinzen needs to order new materials, which causes a delay.

- **Ad hoc day off for production workers.**
  Sometimes materials are not delivered in time, production workers then ask if they can go home because they do not have work. They go home and then the next day the materials arrive. This creates twice the amount of work on the next day.

- **Mistakes in order picking**
  Sometimes the warehouse workers make a mistake with order picking. It causes a lot of time to return the item back in stock and picking the right item.

- **The client wants to change the product (during production)**
  It happens that a client wants to change their product when it is already in production. This causes problems in production because this means they must use different bills of

materials. This is also the case when clients want adaptions to the product before production. This causes a lot of unplanned extra work.

- **Projects wait with entering sales-order**
  Projects only enter the orders in the system once every two days. This causes already a delay. The order is already in the company, but only projects knows about it since it is not yet in the system. This causes a lot of problems because planning knows too late about an order.

- **Not noticing a problem until fine-tuning**
  Problems are not noticed until fine-tuning. It often happens that during fine-tuning there is found a problem in the product. The production workers do not notice the problem during the production process. It cost a lot of time breaking down the machine and then building it back up. This problem goes hand-in-hand with the problem of the production workers being not technical enough.

- **Not clear what score a machine should get to pass the crack test.**
  Prinzen has a so-called Crack-test. It is a way to test how sensitive the machine is to an egg. At first, Prinzen said that the machine should score a ten, but then almost no machine passed the test. So they change the passing mark, but it is not clear what that passing mark exactly is, and how the passing mark reflects the reality.

- **Prinzen is not flexible.**
  Prinzen cut their storage by half two years ago and started working with stock orders. Therefore everything that is in storage is already matched with a certain product. So if there comes a last-minute order, they cannot produce it in time.

- **No insight in current throughput times**
  Prinzen does not have a clear KPI or graph in which the current throughput time is shown. So they know it is too long, but do not know how much too long. This is also because it varies a lot. They cannot see in which department of the company the long throughput time originates.

- **The throughput time varies a lot.**
  From one week to another week the throughput time can change a lot. For projects, this is very difficult. First, a customer gets a delivery time of 5 weeks, then when the customer orders next week the delivery time can be 5,5 weeks. For a customer which already ordered the chickens, this can be a big problem. If the chickens are back in the stable they cannot put a machine in it.

The analysis of initial interviews revealed the issues the processes and issues the company experiences, which were further converted into a conceptual model showing the influences, relations, and causes.

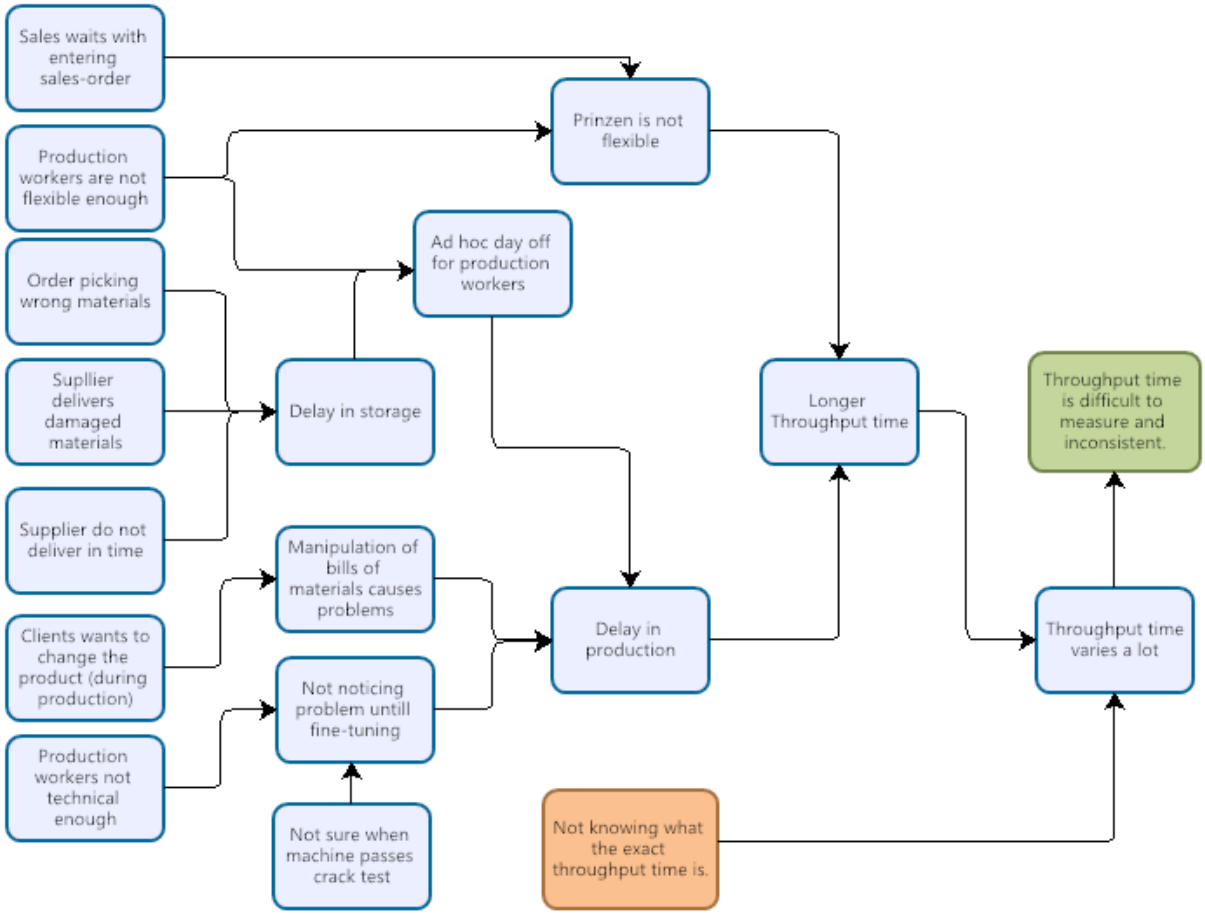


Figure 1 Problem cluster with relations of Prinzen

In figure 1 the problem cluster of Prinzen is shown. The cluster map shows that the throughput time can be affected by a lot of problems. In order to know which problem to solve, one should get insight into what currently the exact throughput time is per department of the company.

Core problem

Prinzen does not have a clear image of what currently the exact throughput time is. It is complex to keep track and optimize processes influenced by multiple dynamic variables.

Motivation

It is necessary to know where the problem is occurring, to solve the problem of the inconsistent throughput time which is difficult to measure. So there needs to be data enabling measuring what the current throughput time is. This is not the case at Prinzen. Choosing this problem the rules of thumb (Heerkens & Van Winden, 2017) needed to be used. The first rule is that the problem cluster shows all problems identified at the company. The second rule deals with following the problems back to other problems which have no direct cause. The core problem must be influenceable. The last rule of thumb is to choose the most important problem if more than one problem remains. Following those rules, a core problem was found to be that Prinzen does not have a clear image of what currently the exact throughput time is. Starting with another problem is not a wise way to go because there was no certainty if it could be the cause of the inconsistent throughput time. The problem is influenceable because one can make sure that data shows what the current throughput time is per department. Projects will be able to give an adequate delivery time if the current throughput time is shown. Throughput time was chosen because it can influence the delivery time, and because it has the biggest impact on

the process, in addition, it is a variable that can include multiple other linked to throughput metrics which, even though may not be known at this stage, can be derived during the thesis work.

1.3. Research design

The following section gives a brief description of the research. The limitations and the intended deliverables are also described in this section.

1.3.1. Research questions

The research is done according to the seven phases of Managerial Problem-Solving Method (Heerkens & Winden, 2021). The goal of the research is to answer the main question. The main question will be answered by researching the knowledge questions.

Main research question

The core problem should be solved to achieve the norm, which is a clear insight into the current throughput time. Based on that information recommendations can be done to Prinzen for achieving their desired throughput time of 4, 8, or 12 weeks. The following main research question should be answered:

How can Prinzen get a clear image of what the current throughput time is concerning the planned throughput figures?

Sub-questions:

- What is the desired process flow of Prinzen?
- What does the current data tell about the throughput time?
- What method can be designed, based on literature, to implement at Prinzen?
  - What solutions design can work for the company to track the processes effectively, measured by the timely discovery of issues and ease of use?
- To which extend does the design satisfy the requirements measured by its' capability to allow for issues detection earlier during the process?

Sub-question 1 From customer order to delivery, every step should be considered. This question provides information to answer the other sub-questions, but also the main question. Based on the desired process the current process is evaluated. The departments and activities that impact the throughput times are known. The desired process flow is made based on interviews. There also is a movie that shows all the departments in Prinzen, this movie served as information for the process flow.

Sub-question 2 Data that is already available should be investigated. For a KPI that will present the throughput time per department, the machine should be tracked all along the process. This question will be answered with a combination of data analysis methods and if possible, process mining. Process mining is only possible when the data is event-based.

Sub-question 3 The core problem is solved by means of a certain method. This method is based on literature and information given by the employees. The method will lead to possible solutions, or the method can be the solution. It is important that the method is easy to use and enables early issue detection. This question provides a redesigned method which forms the basis of the solutions that will achieve the main goal of the research. Extended with excel data analysis, which will give insights into the current throughput time, the designed method discovers issues timely and is easy to use.

Sub-question 4 Evaluating the design is important. For the evaluation, the following questions will be targeted. 1.Does it detect issues early? Evaluation is one of the most important steps in the research. The evaluation should also answer question two. To what extend the designed method matches the reality at Prinzen. The answer to this question is used to form a conclusion of the research. The aim is also to evaluate the solution that is chosen. Interviews with the stakeholders will provide the information for the evaluation and conclusion. The stakeholders need to assess the design on the ability to detect issues early.

1.3.2. Reliability, Validity & Limitations

During the research, there may occur some issues regarding reliability, validity, and limitations. These issues are discussed below, furthermore it also states how to preserve the highest reliability and validity.

The data received from Prinzen may have quality issues.

It is known that the production workers shuffle hours from one project to another. Therefore it is not clear how much time is exactly spent on a certain order. Sometimes an order is finished faster than the data says because the production workers add hours from another order. This is an already known problem. In order to achieve the highest reliability and validity, the data should always be questioned. Also, the production workers should be told that only when the data is of sufficient quality (e.g. not corrupt), the problem can be fixed.

Theories may not apply directly to the reality at Prinzen.

Most of the time theories are limited to the concepts, relationships and, rationale behind phenomena and do not provide ready answers to real-world cases with respect to their applications in practice. The assumptions should always be checked if they apply and are experienced also for Prinzen. To preserve the validity and reliability of the research, the findings from the literature should be narrowed down to the context of the company case, tested, evaluated, and refined with regard to the context of Prinzen.

Interviews are snapshots.

The answer to questions can change from day to day. Answers can also differ on the mood of the interviewee. The department projects could have had difficulties with the production department on the day the interview with projects employee is conducted. The projects employee will then automatically be more negative towards the production department. That is why interviews should be conducted across the whole organization on different days. The results of an interview should not be approached as automatically true.

The covid-19 pandemic effects.

The government of the Netherlands made rules to fight covid-19. One of these rules requires that workers should work as much as possible from home. Therefore employees of Prinzen work remotely. Interviews therefore also should mostly be conducted online. During an online interview, it is sometimes more difficult to get tangible insights about a certain topic of interest. In a live interview, there are fewer limitations, e.g. time-related, formality instead of friendly talks, i.e. extra opportunities to find what the interviewee can also tell more about the topic. Another limitation of the covid-19 rules is that I am not able to see the company working at full capacity and get familiar with the processes on spot.

1.3.3. Intended deliverables

The following section gives the intended deliverables, these are the results of the thesis assignment. Each sub-question provides at least one intended deliverable.

- A process flow of the desired process, which shows the fraction of the time a department can spend on an order.
- Insights in the data that is available of the process at Prinzen.
- Recommendations on data collection.
- A method in which the effectivity of the process is measured based on throughput time.
- Evaluation of the design based on the requirements.

2. Desired process

This chapter describes the analysis of the desired process. Also known as the “soll” situation. The sub-question answered in this chapter is:

What is the desired process flow of Prinzen?

2.1. Process flow chart

The first step in answering the sub-question is to understand the flow of orders in the process. According to Barjis (2008), technical systems fail because the system does not adequately support the underlying business process, this is due to poor conceptual modeling. Therefore the first step is to make a general process flow, and after that for every phase a detailed process flow. The general process flow is only used to determine the phases.

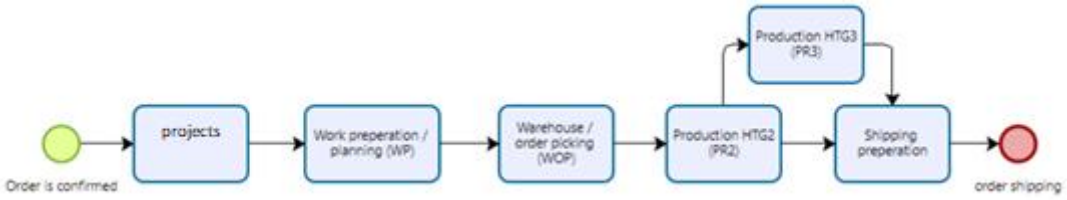


Figure 2 The general process flow of Prinzen

This process flow is based on previously made process flows and conversations with heads of different departments. The outcome of the conversations improved the accuracy of the old process flow. Although the process flow is checked on representing the reality, there are some possible errors. According to Dietz, Barjis, & Goldkuhl, (2002) process flows can be irreproducible and ambiguous. This means that based on the conversations another person will come up with a different process flow, or has the same flow but has a different interpretation of it. To prevent this, the process flow is checked multiple times with different persons in the organization. The process starts from the moment an order is confirmed. Projects puts the order in the ERP system. The order then goes to work preparation/planning. They make the bills of materials and plan the order. The order is then on hold until it reaches its start date in planning. The order is released which then goes to the warehouse. All the materials are picked. Then the order moves to production HTG2, HTG2 makes the machines. If the order is a whole project the order goes further to production HTG3, HTG3 combines the machines into big projects. Then the order goes to shipping preparation. After that, the order is shipped to the customer. There are certain phases not included in the process flow, like shipping the product, or the negotiations with the customer. The pertinent phases are shown in the general process flow.

From the general process flow six phases are determined. Each phase is described in more detail based on information gathered during the interviews. The interview can be found in

appendix E. It is important to understand each step taken in the process, because from that target times for certain tasks or phases are determined.

Projects - this is the department of Prinzen that receives the order from the dealer. Projects is responsible for order entry, with accessory information, in the ERP system. They form a layout of the order if that is necessary. The following phase is able to start working when all the information is in the ERP system. The throughput time of this phase is measured from the moment they receive an order, till the complete order is in the ERP system. Two representatives of projects stated that projects should process an order in two days.

Work preparation - the first step they take is to check the information they received from projects. A plan order is linked to the customer order and becomes a production order. They manipulate the bills of materials according to the wishes of the customer. They also check whether the materials are in stock. The last step is to make hard copies of the bills of materials and a production scheme, these are put together in a map and put aside till the order is released. The second phase starts from the moment the order is in the ERP system till the moment that work preparation has gathered all the information production and warehouse employees need, and work preparation put the order in an office cabinet. A representative of work preparation stated that he processes an order in one day, based on the most standard order. The bigger or more complicated the order is, the larger the throughput time will be.

Warehouse / Order picking - in between the previous phase and this phase the affairs and resource officer must release the order. In other words, there is a possible waiting time between these two phases. To make sure the waiting time is not taken into account with the throughput time. This phase throughput time will start from the moment the order is released. Based on the bills of materials, the employees pick certain materials. The materials will then be put on one cart, such that all materials for a certain machine are on the same cart. The cart is then ready for production HTG2, and the throughput time will end from this moment. The team leader operations, who is also responsible for this phase, stated that the warehouse should pick an order in two days.

Production HTG2 - the production workers need certain things to assemble a machine, the cart from the warehouse and the map with all information from work preparation. In the map, there is a manual that tells step by step how the machine should be assembled. The manual is straightforward, almost everybody is able to build a machine with it. The carts with the materials have wheels and for the machine the production worker gets trolleys. For this reason, the production worker is able to assemble a machine everywhere on the production floor. As a result of that, there is a lot of space available for assembling machines, so Prinzen can make more machines at once. A finished machine goes either to the parking place until it goes to production HTG3 or goes to logistics. Before the machine is finished it is fine-tuned and it needs to pass the crack test. This phase starts when the cart is ready and ends when the machine is finished. The team leader operations stated that this phase should be done in about two to four weeks depending on the size of the order.

Production HTG3 - the fifth phase starts from the moment that the machines are finished at HTG2. Production HTG3 will then combine more machines in bigger projects. The belts between different machines to transport the eggs are made. The complete project is fine-tuned, and also needs to pass the crack test. The phase is finished if the complete project is finished and ready to be packed and shipped. Production HTG3 will only work on an order if more

machines need to be combined. The team leader operations stated that production HTG3 should finish an order in one to four weeks again depending on the size of the order.

Shipping preparation - this phase starts either when a machine is finished or when a project is finished. The logistic employee puts the machine on a pallet and makes sure it can not move. At last, a manual is added and the machine is covered in plastic. Then a truck comes to pick up the machine. From that moment this phase is finished, and the whole throughput process is finished. This phase should at most take two days to process an order, according to the team leader operations.

2.2.Order classification

The desired process flow processes orders in 4, 8, or 12 weeks. However, in the ERP system this classification cannot be found. This brings up the question: “How do employees know which classification an order or machine gets?” In the interview, employees have to classify the machines and certain orders from the ERP system. The eggficiency plan with the 4, 8, or 12 weeks classification should be widely supported in the whole organization. The answers to the questions tell whether this is true or not.

The results of the interview questions about the order classification can be found in appendix F. The tables show the role of the interviewee on the horizontal header and the machines on the vertical header. In the cells, the throughput time in weeks can be found. To test whether every employee has the same idea about the order classification the answers were compared to each other.

From the results, the following summary can be concluded. Only 5 of the 16 machines that Prinzen sells have been classified with the same number by all interviewees. 6 machines are classified by everyone but one employee with the same number. Below the results for four machines are shown, because the employees have a different view on the throughput time for that certain machine.

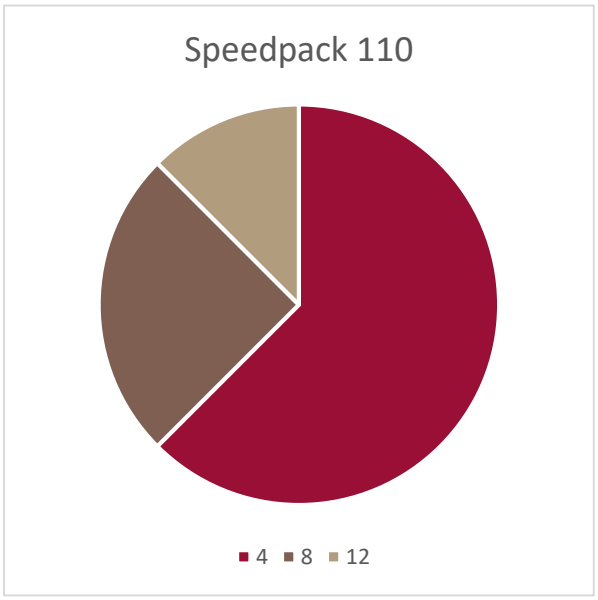


Figure 3 Speedpack 110 classification

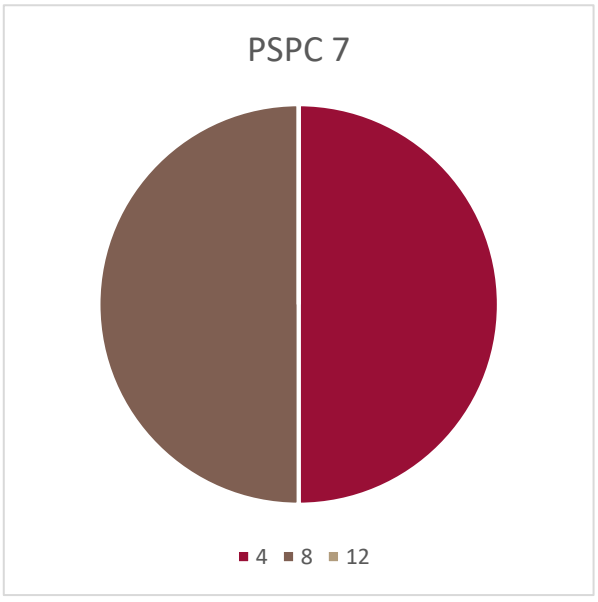


Figure 4 PSPC 7 classification

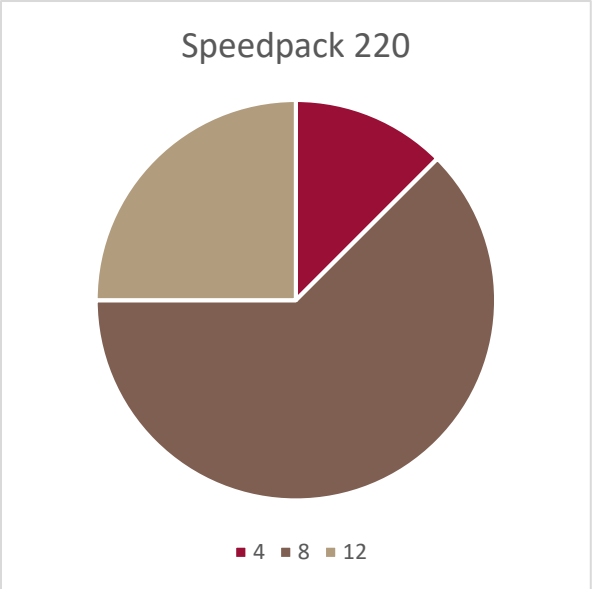


Figure 6 Speedpack 220 classification

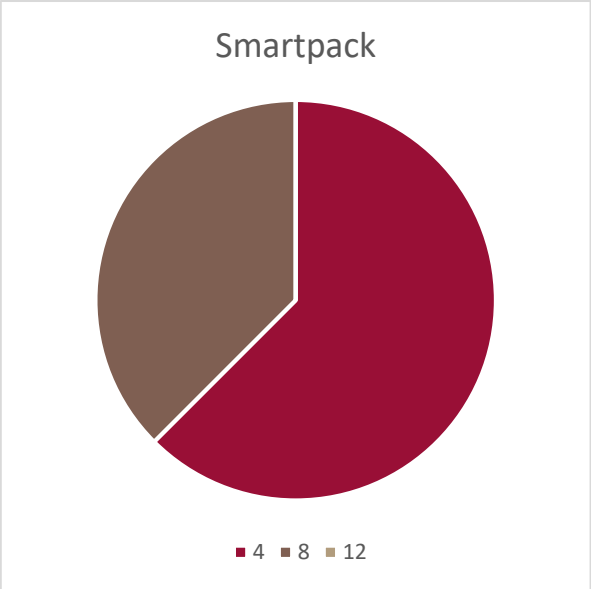


Figure 5 Smartpack classification

The above figures show that not for every employee the eggfiecency plan with the idea of 4, 8, and 12 weeks is clear. For example the PSPC 7, half of the interviewees say the throughput time should be four weeks, the other half says eight weeks. A solution should be found to make sure that every employee knows the exact amount of time in which the machine should leave the company. Orders consist most of the time out of multiple machines, therefore the same question was asked for eight example orders. Out of the eight orders, four are classified in the same category by all interviewees. Two orders are, similarly to the machines, classified in different categories and therefore shown below.

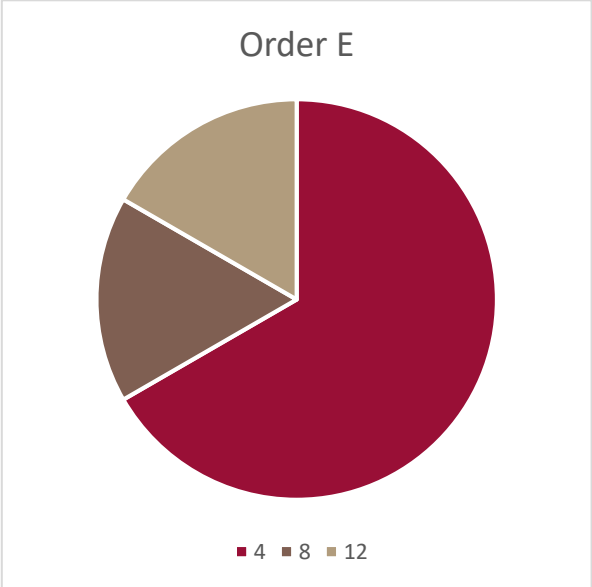


Figure 7 Order E classification

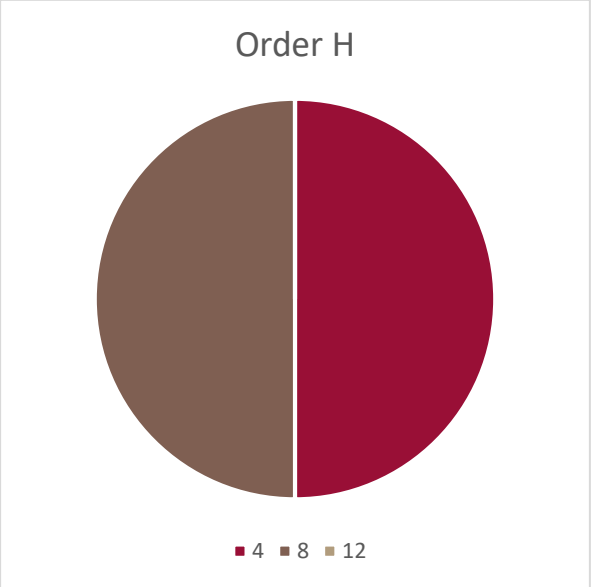


Figure 8 Order H classification

Half of the interviewees think that order H has a throughput time of four weeks, the other half thinks it is 8 weeks. There can not be any difference in what the target time is for a certain order. This difference will cause difficulties in analyzing the throughput time. Therefore in the method a solution for this problem will be taken into account.

2.3. Process conclusion

The general process flow of Prinzen is described in Figure 2 The general process flow of Prinzen. The process flow can be divided into six phases. The throughput time will be measured per phase in such a way that the problem owner, the team leader operations, is able to detect a problem in a certain phase early.

The desired throughput time is 4, 8, or 12 weeks according to the complexity and size of the order. The research in this chapter showed that not all employees classify the orders and machines in the same way. This causes difficulties in analyzing the data. For example, Prinzen receives a certain order, projects classifies the order as 4 weeks while the team leader operations thinks it is 8 weeks. The order will eventually flow through all the phases in 6 weeks. Projects will then think they did a bad job while the team leader operations thinks he did a great job. All employees must be on the same page. Therefore I recommend adding a label to an order which tells the category the order has, 4, 8, or 12 weeks. This recommendation will be used in the dummy data during the research, such that it makes it easier to analyze the throughput time.

3. Data analysis

In this section, an overview of the collected data is given. In this overview, the place where the data is stored is mentioned. Then the data overview is analyzed, and sub-question two is answered.

What does the current data tell about the throughput time?

The throughput time is taken as a metric because it describes a source of competitive advantage and the fundamental measure of manufacturing performance. (Neely, Gregory, & Platts, 1995). The metric tells something about the efficiency of Prinzen. According to Neely et al. (1995) efficiency measures the economic utilization of the firm's resources. The data however should be suitable for certain data analysis methods. This is also taken into account during the data collection.

3.1.SAP (ERP system)

SAP is the Enterprise Resource Planning system used at Prinzen. SAP ERP is used within the Vencomatic Group. The system is suitable for large to middle-large companies. SAP is a very complicated system. Only a handful of people tend to have all the knowledge on how to use SAP. The search for data started with them, with an event log of the process being the goal. Despite being an ERP system that is able to track orders, Prinzen does not do this. Therefore the employees are not able to hand over an event log. The system however will be used to track the order input date. This is a data type which is kept track of in SAP. Another type of data that is valuable for the research, is the wished delivery time by the customer. These two things are taken into account in the data analysis.

3.2.Monday (project management tool)

Monday is a project management tool. In Monday every employee can be assigned to a certain task. Each manager is able to see what the employees are working on. Also, every order can be tracked in Monday. Every department is able to work in Monday. Employees are able to communicate with each other via Monday. They can start a discussion linked to an order. Managers are able to see in which department an order is delayed. In other words, Monday is a good alternative to get data from.

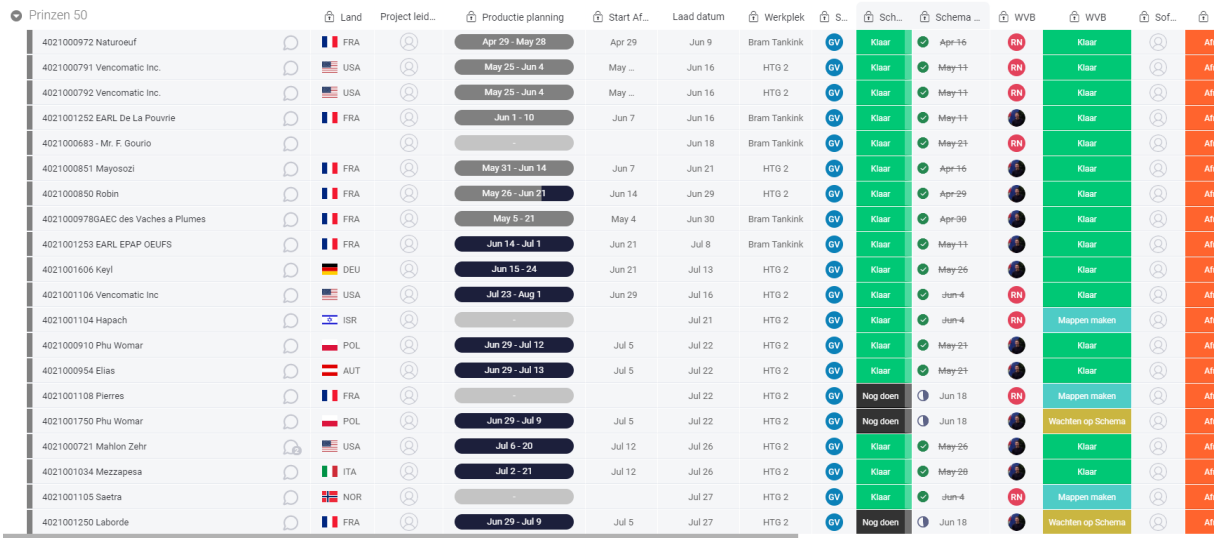


Figure 9 Order dashboard in Monday

In the image above an overview of a part of the order dashboard is shown. For every machine, in this case, the “Prinzen 50”, a separate list keeps track of the orders. In the first column the order number is shown with the buyer. The second column shows the country the order goes to. The fourth column shows the production planning, and the fifth column shows the start of assembly. Then the loading date is shown in the sixth column. The following five columns are for work preparation. After that there are some columns for software and manual making. The dashboard shows dates for work preparation tasks and production. This is useful data to implement in the data analysis. However similar to the ERP system data cannot be withdrawn in the form of an event log.

3.3. Data analysis conclusion

This section answers the research question: What does the current data tell about the throughput time. In this chapter, the search for data about the throughput time is done in several systems. However, in all the systems there is not any type of data that tells what the current throughput time is. Prinzen knows their throughput time is too long from feeling rather than from data telling them. This makes it difficult to research the throughput time, since for a dashboard with KPIs about the throughput time the data is also necessary. For the KPIs data should therefore be created, synthetic or dummy data is going to tell what type of data Prinzen needs to store in their systems. Besides that, the dummy data is also going to show what can be done with that data if it is available. However, currently Prinzen does not have the data available to say anything about the throughput time. The assumption that the throughput time is too long is only based on feelings from the practice.

4. Literature

4.1.Throughput time

A quick throughput time is getting more and more important for companies. “Leading companies seek to introduce products quickly and to respond to customer orders in minimum time” (Lederer & Li, 1997). Order today, receive tomorrow is the standard for the new generations. In the case of Prinzen, a company with a make-to-order (MTO) strategy, the delivery time depends on the throughput time plus the shipping time. The shipping time varies because of the distance between Prinzen and the client. Consequently, a decrease in delivery

time requires a shorter throughput time. Quick throughput time leads to improvements in cost, productivity, and profitability (Stalk, 1988; Stalk & Hout, 1990; Schmenner, 1988; Blackburn, 1991; Suri & De Treville, 1986; Suri, 1994). Tatsiopoulos, & Kingsman, (1983) state that the quality of the planning data affects the production planning. Good quality of engineering data (e.g. bills of materials), capacity data (e.g. workforce), and lead time data (e.g. manufacturing lead times) are all required for successful production planning according to Tatsiopoulos, & Kingsman, (1983). From the problem cluster (figure 1) it can be stated that Prinzen has more problems that lead to unsuccessful production planning. The bill of materials lacks quality and the insights in the lead time are not present. Insights in the lead time are necessary to shorten the throughput time.

4.2.Process mining

One of the possible ways to come to an insight in the throughput time is to apply process mining. Searching theories about process mining is done according to a systematic literature review. Process mining analyses the event logs of a process (Cheng & Kumar, 2015). Process mining aims to discover, monitor, and improve an existing process (Tadanai & Tanuska, 2015). Therefore process mining is the ideal method to analyse and improve the throughput time of Prinzen. The only counter side of process mining is that the event logs need to be very clear. Noisy event logs produce bad models (Cheng & Kumar, 2015). There are multiple software options for process mining, for instance, Disco and Prom. Prom is the most commonly used one (Grigorova, Malysheva & Bobrovskiy, 2017). The first step is to investigate if the data at the company is sufficient enough for process mining, otherwise, the research should be continued into other data analysis methods. The team leader operations of Prinzen has already informed that process mining would certainly be a preference, but he also noticed that the data is not sufficient yet. The level of detail in the data covering all aspects of the process is not sufficient enough. Therefore the data needs to be generated or other methods of solving this problem need to be found. This can be a framework that would help the company to collect relevant data to enable process mining.

4.3.KPI's for process performance

Another way to analyze the process performance of Prinzen is to create several key performance indicators (KPIs). KPIs can reveal poor or satisfactory performance and improvement potential (Lindberg, Tan, Yan, & Starfelt, 2015). The KPIs can be put in a dashboard such that the COO of Prinzen can easily detect how the process performs. According to Lindberg, Tan, Yan, & Starfelt, (2015) important KPIs for operation are scheduled operation time in percentage and the actual uptime of the scheduled time in a percentage both over a certain period. A useful planning KPI is the planned time – actual production time. This KPI shows if the planning is too tight. KPIs related to time are meaningful for this problem. "Time has been described as both a source of competitive advantage and the fundamental measure of manufacturing performance" (Neely, Gregory, & Platts, 1995). Certain KPIs mentioned by Neely, Gregory, & Platts (1995) that may be of value for this research are value-added time and distance traveled. The value-added time is measured as a percentage of the total time. Maskell (1989) suggests seven principles for designing a performance measurement system.

- The KPIs should directly contribute to the firm's strategy
- The KPIs should not only measure financial values

- There is not a KPI that suits all departments, therefore KPIs should differ per department.
- The KPIs should change as the process is changing.
- The KPIs should be easy to use
- The KPIs should provide fast evaluation
- The KPIs are designed to stimulate improving the process, rather than monitoring the process.

Shahin, & Mahbod, (2007) propose that KPIs should be based on the criteria of SMART (Specific, Measurable, Attainable, Realistic, and Time-sensitive) see Figure 10 Smart model.

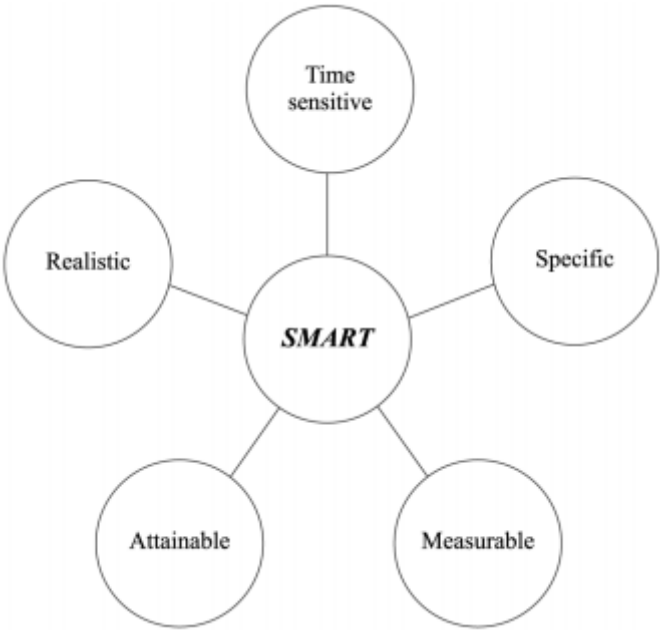


Figure 10 Smart model

The key performance indicators should be as specific as possible to make them measurable, e.g. not too broad or vague. KPIs are measured against the desired situation, this situation must not be out of context reach. Hence the KPI should be attainable. Furthermore, an attainable KPI should also be realistic. With a certain KPI comes a goal. The goal must be time-sensitive, in other words in a certain amount of time the goal must be completed.

4.4.Operationalization

The conclusion from the literature is that there are several ways to improve throughput time. Process mining is one of the methods that can be used to give an inside into where the variable throughput time is caused. Also, process mining should give a clear image of what the throughput time is. The contra-side to using process mining is that the data should describe every event in the process in detail, and the data should be available. Prinzen does not yet have the right data to perform process mining, further research at the company will reveal if process mining is available after creating or changing the data. If the outcome of the research is that process mining is not possible, then KPIs should be formed to give a clear image of what the throughput time is. The KPIs will be drafted according to qualitative research and quantitative research. The qualitative research will be interviews with the stakeholders, they will state what is necessary to know for improving the throughput time. The quantitative research will be a search into the data that is available at Prinzen. Possible KPI's that can be

created according to the data will be made. To conclude the method that will be chosen is based on the availability and the detail of the data.

5. Solution method

In this chapter, the solution method is described. The chapter answers to the following research questions are provided:

What method can be designed, based on literature, to implement at Prinzen?

&

What solutions design can work for the company to track the processes effectively, measured by the timely discovery of issues and ease of use?

The methods and theories found in the literature are combined into one method, such that the method gives results and a solution for the core problem at Prinzen. The methods are redesigned to be applicable to this company case. The literature found in chapter 4 is also used for the method construction.

5.1. Prerequisites

The method has some prerequisites, these are discussed in this section. Literature may be limited to the concepts, relationships and, rationale behind phenomena. The methods in the literature also do not provide ready answers to the company case. To make sure the method provides a solution for this problem of Prinzen a list with prerequisites is made.

Ease of use – The problem owner, the team leader operations, must be able to use the method that is designed in this chapter. All employees are able to work with the solution that will result from this method. The team leader operations or any other employee of Prinzen should be able to redo the steps taken in the method. This idea should be taken into mind while working on the method.

Characteristics of the case – The method needs to fit the characteristics of the company case. The process is divided into phases, these phases should be taken into account when designing the method. Prinzen is a medium company, with around one hundred employees present on daily basis. The process flow of Prinzen is taken into account with regard to designing the method. Methods are adapted to fit these characteristics of Prinzen.

Measurability – First of all the method is able to measure the throughput time of Prinzen, in addition to this, the method is also able to measure the throughput time of the phases in the order flow of Prinzen. Secondly, the method detects issues regarding the throughput time early, and based on that gives a warning to the user of the end product. The method also generates some numerical results, which measure the throughput time of Prinzen and assess it on the target times. The method gives results otherwise there is no value given to the method.

Future value – The method design needs to be applicable in the future. Especially in this case, where there is a lack of data, and dummy data is created. In the future Prinzen is going to gather the missing data, such that they will be able to execute this method with real data. It is known that Prinzen is going to change its ERP system in 2023. After this, the method should still be valuable to Prinzen.

5.2. Steps of the method

The method construction is described in five steps. The steps are described in each section below. Some steps consist out of multiple actions. Figure 11 below shows the steps including the activities of the method.

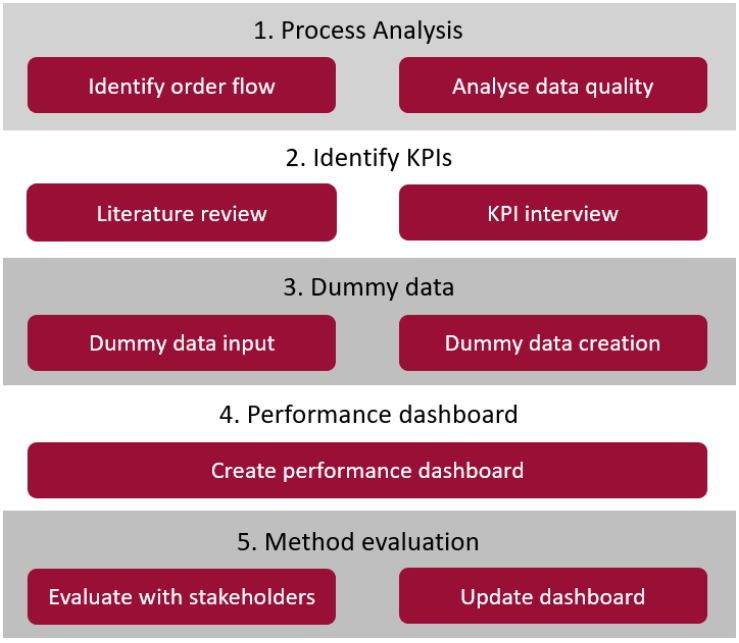


Figure 11 Steps of the method

5.2.1. Step 1 – Process analysis

To get insight into the throughput time first, the process flow of an order must be researched. According to Youngman, (2006) "the primary objective of the process-flow analysis is to optimize available resources." The time to process an order at each step in the process is an element that is researched often with process-flow analysis. (Youngman, 2006). Second the available data must be analyzed. The next steps of the method depend on the quality of the data. According to Huh, Keller, Redman, & Watkins (1990), data quality is important but receives too little attention from companies.

Identify order flow – An analysis of the throughput time starts with identifying every step it takes to process an order. This is done by conducting initial conversations with representatives of the different departments. Provided that a company already has mapped the process flow, the initial conversations will then validate the mapped process flow. If the company does not have a mapped process flow, the initial conversation supports the researcher in making a general process flow. The detailed process flow is described in the text.

The process flow will be divided into phases. At least a phase for each department in the order flow. The start and the end of a phase need to be an activity. These activities are the basis for the event log. Each time a start or end activity of a phase is completed the time and date are noted in the event log. In this particular company case, there is a break in between the process flow, this activity is the order release date activity, i.e. the order has a waiting time before this activity. The waiting time however is not taken into account in the throughput time, since it is a waiting time based on the wishes of the customer. For example, the customer wants to receive the order in ten weeks, the order should have a throughput time of 4 weeks, then the order will

be in a cabinet for at most six weeks based on planning. Certain activities like this also are registered in the event log. The event log will form the basis for analyzing the throughput time.

Analyze data quality – some of the results of the method are numerical, the research needs input for these results. Since this is a company case the input will be data available at the company. There are different systems that can be used as input for the research. The ERP system is the most important of those. For this company case, there was also some data in Monday which is a planning tool. This can be different for every company case, but all the data that is relevant for the research is gathered. The data is assessed on its quality e.g. detecting the conformance with SMART rules, missing data, incomplete data, errors, format mismatches, etc, conformance with event log format such as activity, timestamp, actors, target. Then a method like process mining will easily give results about the process flow, e.g. visualizing process discovery maps, validating a process for compliance with the desired processes, bottlenecks, process statistics, etc. The processing or creating of data will be done in further steps of the method.

5.2.2. Step 2 – Identify KPIs

The results of the research are shown with KPIs, therefore a list is conducted with KPIs. The list with KPIs is formed in two steps. First, a literature review about possible KPIs is conducted, second an interview with stakeholders to expand the list and to select certain KPIs from the literature.

Literature review – in different articles KPIs that can be of value for this research are searched. The literature review is done in different databases, in such a way that the largest amount of KPIs is found. Different search terms are used, but they will all be in the line of this research. The research will use KPI because the KPIs serve the first relevant target for identifying issues and because the analysis should also start with the more reliable collection of data to enable analysis from different perspectives (Ferreira, Silva, Casais, Pereira, & Ferreira, 2019). For this company case, the collection of data is not that reliable, data is not available or sometimes corrupt not allowing to obtain insights. KPIs will be to first show how successful a company is on certain goals (Velimirović, Velimirović, & Stanković, 2011). The goal for this company case is to have a throughput time of 4, 8, or 12 weeks.

KPI interview – to match the KPIs with the company, interviews are held with the heads of the different departments. The heads of the departments are able to add KPIs on their own to the list. The KPIs mentioned by the heads of the departments are useful for that certain department, therefore an option is to make a dashboard per department. E.g. the logistic department is not interested in how long the sales department works on one order. If the list is complete with KPIs, the problem owner will next prioritize the KPIs. This will be done with the Analytical Hierarchy Process (AHP) method. This method compares the KPIs in terms of relative relevance. The method results in aggregated weights because the method gives a priority to the KPIs (Hruska, Prusa, & Babic, 2014). The method is used to determine the importance of the KPIs. Since the dashboard must be easy to use not all KPIs will be shown. Then the selection of the KPIs will be based on the prioritization by the problem owner.

5.2.3. Dummy data

The thirds step in the method construction process deals with the data collection. This step is only necessary if there is an absence of compliant data found in step one. This step is necessary for the specific company case of Prinzen, meaning that this will not always be the

case for every company case, as the data available at Prinzen was not compliant enough to get results for the research.

Dummy data input – to make sure the dummy data represent the reality as best as possible, the interview contains detailed questions about the average time spent between the measuring points. Every interviewee is asked how long the works takes in his phase. The problem owner then provides more questions/answers about the process, e.g., questions about the number of orders per day, or the percentage of a certain order type. The information gathered in this step is then used to make dummy data.

Dummy data creation – With the information from the previous step, dummy data is created. With historical data, synthetic or dummy data can be created (Abtew, Moras, & Campbell, 1990). The dummy data represents an event log. With this event log certain KPIs are made. The throughput times in the data are randomly created with a normal distribution. This is the best fitting theoretical distribution (Abtew, Moras, & Campbell, 1990). The throughput time is then added to the end date of the previous phase. Also, with each order an order number and a throughput time classification are generated. The time classification is generated based on the percentages of the real orders time classification. The data is created with Visual Basic for Applications (VBA). This is to make sure that nobody is able to change the code by accidentally deleting a cell in the sheet. Before the KPIs are created, the team leader operations will check the data to make sure it represents the real data.

5.2.4. Step 4 – Performance dashboard

The fourth step in the method construction process aims to create a performance dashboard. According to Neely, Gregory, & Platts (1995), the dashboard measures the effectiveness and efficiency of actions. All of the previous steps are combined into the performance dashboard. The performance dashboard can be defined as “a multi-layered application built on a business intelligence and data integration infrastructure that enables organizations to measure, monitor, and manage performance more effectively” (Eckerson 2006).

The dashboard must be easy to use and warns the user early about issues in the order flow. The dashboard consists of graphical views of the key performance indicators, such that a user is able to monitor those KPIs (Suk, Hwang, Dai, Caldas, & Mulva, 2012). The dashboard will also have some kind of simulation function in such a way that the user is able to validate the effectiveness of the changes to the order process. Overall the dashboard should show for this particular company case what can be done if the data was present, what type of data is needed and what type of insights it can provide to facilitate management level decision making for solutions, e.t. avoiding issues like delays.

5.2.5. Step 5 – Method evaluation

The method should be evaluated with the stakeholders. The results of the method, the performance dashboard, should be valuable input for processing the orders. The dashboard is updated when new data or insights in the process are found. If the process changes the dashboard with the KPIs should change as well, e.g. provide dynamic outputs.

5.3. Research methods

There are two research methods used for this company case. Qualitative research and quantitative research. The interviews held to gather data for the research belong to qualitative research. Qualitative research is the opposite of quantitative research. Qualitative research is about analyzing non-numerical data. In contrast with qualitative research, quantitative research

is about analyzing numerical data. Tracy (2010) states eight criteria for qualitative research: worthy topic, rich rigor, sincerity, credibility, resonance, significant contribution, ethical, and meaningful coherence. Qualitative research is used in the interview to find out important aspects of Prinzen's processes and improvement needs from various perspectives.

In the research quantitative research methods are used. The KPI dashboard and the analysis of the dummy data are quantitatively processed. According to Polit & Beck (2012), Quantitative research uses precise measurement to examine phenomena. “In quantitative studies, the researcher uses standardized questionnaires or experiments to collect numeric data.” (Rutberg, & Bouikidis, 2018). In the interview, data is retrieved quantitatively by asking for numerical data about the throughput times. The solution method described above uses quantitative methods as well as qualitative methods, that is why this solution method is a mixed research method. Tashakkori and Creswell (2007) state that “research in which the researcher collects and analyses data, integrates the findings, and draws inferences using both qualitative and quantitative approaches or methods in a single study or program of inquiry” is a mixed-method research. This solution method is a mixed research method which is using quantitative method, e.g. the interview questions about the process times of orders and qualitative methods, e.g. the interview questions about aspects of the Prinzen processes, at which for the qualitative method the eight criteria of Tracy are kept in mind doing the research.

5.4. Solution method conclusion

In this section, the research questions mentioned at the beginning of the chapter are answered. The method that is designed is a mixed research method. The data from the interviews are part of the qualitative research and the dummy data, dashboard, and KPIs are part of the quantitative research. An overview of the method that is described in this chapter can be found in Figure 11 Steps of the method. The solution design consists out of a dashboard that is based on dummy data. The dashboard contains a function which in some way can simulate different input variables. Therefore the problem owner, the team leader operations, is able to test whether changes in the order process will affect the throughput time.

6. Implementation

In this chapter, the implementation of KPIs in a performance dashboard is described. The KPIs are selected and implemented in a dashboard. The creation of the dummy data is also mentioned in this chapter. The product of this chapter is a dashboard based on dummy data with KPIs selected by the team leader operations.

6.1. Key performance indicators

In this section, the final list of key performance indicators for the KPI dashboard is selected. The choice of KPIs in the list is based on the findings from the literature and the interview with the stakeholders of different departments at Prinzen. From the list then KPIs are selected based on prioritization, e.g. indicated by the problem owner, in this case, the team leader operations.

6.1.1. KPI list

The following list of KPIs is composed of KPIs from the literature and KPIs suggested by the stakeholders in interviews. The list thus consists of a combination of KPIs selected from theories and KPIs selected from Prinzen experiences and practice. The KPIs from the practice are suggested because of a lack of insight into the order process specifically at Prinzen. The

KPIs that are suggested by the employees at Prinzen will thus get a higher prioritization since these are KPIs mentioned by lack of insight in the real-world process situation at Prinzen. In section 6.1.2 the KPIs are prioritized by the team leader operations.

Warning for too long throughput times – The team leader operations would like to see the last orders that did not achieve their throughput time classification. The throughput time of the order will then be divided over the different phases, and each phase is assessed on its phase throughput time.

Planned operation time (in FTE's) - This KPI shows the number of full-time equivalents (FTEs) per department or phase for the upcoming ten weeks. In combination with the KPI below the team leader can use this information for planning orders. If this KPI has a peak in the graph, and the KPI below has a low point in the graph, it means that in that week more orders can be planned.

Planned orders in progress – This KPI shows how many orders are planned in the coming ten weeks per phase. The bigger the number the tighter the planning is. It only shows the orders that have not passed the release date of the order. This is because the KPI needs to function as information for planning. Especially for the short delivery time orders, based on the graph the team leader operations is able to tell when he can plan more orders.

Value-added time (percentage) – The value-added time measures the time an employee of Prinzen actually works on one order. This is then put against the total amount of time it takes to process an order. The reverse of this KPI is the waiting time against the total amount of time it takes to process an order. This is a very useful KPI, but still unfeasible KPI, since for every order the time of each activity must be tracked which is impossible with the current setup at Prinzen as Prinzen does not keep track of these processing times.

Average time per phase – The time an order is at a certain phase. This KPI shows the average time a phase needs to process an order over the last week. The number will be visualized in a key indicator since an order can have a different time classification. So if the orders of last week will be made in exactly the same amount of days as the goal then the key indicator will be achieved.

The number of orders in progress per phase – For the employees of Prinzen it is useful to know the number of orders that are in progress in the phases before them. The team leader operations benefits from this because he is able to shift people around in the production according to that number.

6.1.2. KPI prioritization process

The order of implementation of the KPIs is based on the importance of the KPIs relative to each other. The problem owner, the team leader operations, compared the KPIs in terms of relative relevance. To find the relative importance of the KPIs, the Analytical Hierarchy Process (AHP) method is used. The AHP method uses pairwise comparisons to identify the weights. The leading KPI (row KPI) is given a number between 1 and 9 to show the relative importance of the column KPI. If the KPI in the column is relatively more important than inverse values are given.

	KPIs	1	2	3	4	5	6
1	Warning for too long throughput times	1	3	2	0,25	0,33	0,5

2	Planned operation time	0,33	1	0,5	0,16	0,2	0,25
3	Planned orders in progress	0,5	2	1	0,2	0,25	0,33
4	Value added time	4	6	5	1	2	3
5	Average time per phase	3	5	4	0,5	1	2
6	Number of orders in progress per phase	2	4	3	0,33	0,5	1

Table 1 Relative importance KPIs, team leader operations

The weights are calculated with the following equations. $x_{i,j}$ represents the number in column i row j. $w_{i,j}$ represents the weighted relative importance. The weighted relative importance is measured by the total of one column. This is shown in the following equation;

$$w_{i,j} = \frac{x_{i,j}}{\sum_{i=1}^{i} x_{i,j}}$$

All the values for $w_{i,j}$ are again put into a table. This results in the following table.

	KPI	1	2	3	4	5	6
1	Warning for too long throughput times	0,09	0,14	0,13	0,08	0,08	0,07
2	Planned operation time	0,03	0,05	0,03	0,07	0,05	0,04
3	Planned orders in progress	0,05	0,10	0,06	0,08	0,06	0,05
4	Value added time	0,37	0,29	0,32	0,41	0,47	0,42
5	Average time per phase	0,28	0,24	0,26	0,20	0,23	0,28
6	Number of orders in progress per phase	0,18	0,19	0,19	0,14	0,12	0,14

Table 2 Weighted relative importance KPIs

Then the weights of the criteria, the KPIs, are calculated with the following equation. In this equation K_i represents the weight of KPI i.

$$K_i = \sum_{j=1}^{j} w_{i,j}$$

The weight, or in other words the importance, of the KPIs can be found in the table below. The KPIs are in section 6.3 implemented into a dashboard. However, another important criterion is that the dashboard must be easy to use. (Sedrakyan, Leony, Muñoz-Merino, Kloos, & Verbert, 2017). The KPIs will be implemented based on their priorities, i.e. from the most important to least important. If the dashboard is too complex then KPIs will be abstracted away to preserve the easiness of use, which can be viewed by further drill down, filter, or zoom in/out techniques (Sedrakyan, Mannens, & Verbert, 2019).

KPI	Weight
Warning for too long throughput times	0,59
Actual operation time	0,27
Planned orders in progress	0,40
Value-added time	2,28
Average time per phase	1,49
Number of orders in progress per phase	0,96

Table 3 KPI weights

The weights calculated from the pairwise comparison done by the team leader operations results in the following list of KPIs from most important to least important.

- Value-added time
- Average time per phase
- Number of orders in progress per phase
- Warning for too long throughput times
- Planned orders in progress
- Planned operation time

Since value-added time is yet not able to be implemented in the dashboard, I advise the company to gather data to make sure this KPI can be implemented in the future. In the dashboard, the other KPIs will be implemented. It can be that the least important KPIs are abstracted away from the dashboard if the information made visible on it will get too complex. More information about the dashboard can be found in section 6.3.

6.2. Data selection procedure

In this section, the procedure of the dummy data creation, and data selection procedure are described. The dummy data is used as input to show Prinzen how they can get an insight into their throughput time, but moreover insight into the throughput process. For the dummy data, some information about process data was used as input to generate dummy data that will represent the unavailable data as much as possible.

6.2.1. Dummy data input

The input for the dummy data came from interviews with the stakeholders of different departments of Prinzen. Each interviewee was asked to fill in a table in which they had to put the maximal process time of the orders per phase. The other necessary information for generating the dummy data is given by the team leader operations, e.g. the distribution of the orders being 60% orders with the 4-week classification, 30% orders with 8-week classification, and 10% with the 12-week classification.

The generation of the dummy data is built in excel with the use of VBA. The input data is fixed however the sheet is built such that the user of the dummy data generation tool is able to experiment with the data, in such a way that the data fits the real-world data as best as possible. For Prinzen the team leader operations is able to change certain data based on changes he makes in the process. He is able to change the number of full-time equivalents, this will affect the dummy data about the capacity. The dummy data input sheet can be found below.

Start date:	7-7-2021
Date today:	21-10-2021

	Average	Standard Deviation	Minimum	Maximum
Number orders per day	3	2	1	5
Projects Time	1	0,2	1	3
Work Preperation Time	1,2	0,2	1	4
Warehouse Time	2	0,01	2	3
HTG2 Time (4 weeks)	12	1	10	14
HGT2 Time (8 weeks)	17,5	1,75	14	21
HGT2 Time (12 weeks)	21	3,5	14	28
HTG3 Time (8 weeks)	7	1	5	14
HTG3 Time (12 weeks)	17,5	3,5	7	28
Order Finish Time	2	1	2	3

Number of FTE's	Average	Standard Deviation	Minimum	Maximum
Projects	5	0,1	2	5
Work prep.	4	0,1	2	4
Warehouse	10	1	6	11
HTG2	50	5	40	60
HTG3	30	5	20	35
Shipping prep.	3	0,1	2	4

Figure 12 Dummy data input sheet

6.2.2. Dummy data macro's

Three VBA macros are created to generate the dummy data for Prinzen. The first macro generates the event log data, the VBA code for this macro can be found in Appendix G.1: macro generating event log data. The macro generates event log data from the past. The macro can be described in eight major steps, irrespective of the variable generating and getting the input from the input sheet.

- Generate basic order info, such as order number, order class, order entry date, and wished delivery date.
- Generate end projects date
- Generate end work preparation date
- Generate order release date
- Generate end warehouse date
- Generate end production HTG2 date
- Generate end production HTG3 date if necessary. (4 weeks order does not go to HTG3)
- Generate order finish date

For each step, a processing time is calculated from a normal distribution with the input variables for that certain step. Then the time is added to the previous end date, which is the start date for this phase. Except for the warehouse phase, this phase starts with the order release date. Before the end date is put into the event log sheet, the macro checks whether the date is in the weekend or not. If the date is in the weekend the macro will add one or two days, depending on the day of the weekend. Then the end date is written in the event log sheet.

The second macro, which is called “Capacityhours”, generates per week the number of full-time employees that is available at each phase. The macro loops over all days and starts with checking which day the start day is, e.g. Monday, Tuesday, etc. This is because the macro apart from the first day, only generated the data if the day is a Monday, which is the start of the week. Each Monday it writes the week number, the start and end day of the week. The number of full-time employees is calculated with a normal distribution with the input data from the input sheet. On the first day, it generates the data for the first week, this does not depend on the type of day. The VBA code for this macro can be found in Appendix G.2: macro generating capacity data.

The last macro enables the user of the sheet to simulate one day further. The user, in this case, the team leader operations, is able to change certain input data, and then simulate how those changes affect the order process. The macro code can be found in Appendix G.3: macro simulate one day. The macro code can be divided into two parts. The first part is to update the capacity sheet, and the second part is to update the event log sheet.

The first part of the macro is not that different from the macro that generates the past capacity data. The method first calculates how many weeks already have been past. The next step is the same as the initial code of the ‘capacityhours’ macro, the macro checks whether the extra simulated day is a Monday, if it is it generates another week of capacity, otherwise this part of the macro will be skipped.

The second part of the macro is different from the initial code for generating the event log data. The code loops over all rows, when one row represents one order. The code then searches the last column of that row. Based on the last row the macro calculates whether the order is already in the next phase or not. If the order is in the next phase, in other words, the start date of the phase plus the order processing time is smaller than the date of today. If that is the case the macro will write the date of today in the next cell of the row. If the macro reaches the end, so the last order that was already in the event log is gone through the macro, the macro generates a random amount of orders and writes them in the cells below.

6.2.3. Dummy data output

The output of the dummy data generation macro is an event log and a table in which the capacity is shown per week. The most important of these two is the event log. Each row in the event log represents an order. The image below shows one complete order in the event log.

OrderNr	OrderClass	WishedDeliveryDate	OrderEntryDate	EndProjectsDate	EndWorkprepDate	OrderReleaseDate	EndWarehouseDate	EndHTG2Date	EndHTG3Date	OrderFinishDate
401	4	10-9-2021	7-7-2021	8-7-2021	9-7-2021	23-8-2021	26-8-2021	7-9-2021		9-9-2021

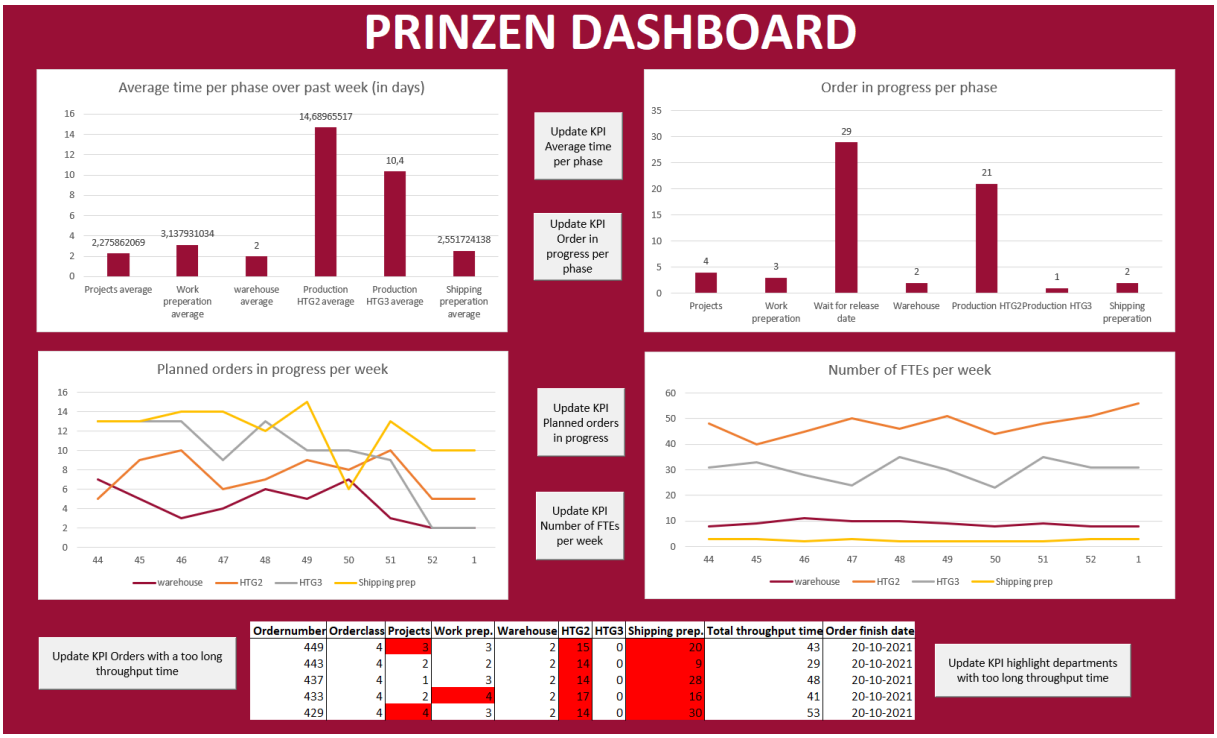
Figure 13 Event log example of one order

The event log table is currently not available at Prinzen, however this data is necessary for the KPIs that track the order process. As stated in chapter 2.3 the order label is added to the event log. This label, the order class, is necessary to tell how Prinzen performs on their own goals. The other cells in the event log represent either the start or the end of a certain phase. Prinzen should keep track of the process in an event log that more or less looks like this event log. Based on that event log the problem owner, in this case, the team leader operations is able to track issues and orders in the process.

6.3. KPI dashboard

The dashboard shows five example KPIs that can be implemented with the type of data gathered in the chapter above. The KPI with the highest prioritization has not been

implemented since this data was not available. The other five KPIs were implemented. The dashboard is shown in the following image, the sections below the image explain the parts of the dashboard.



Ordernumber	Orderclass	Projects	Work prep.	Warehouse	HTG2	HTG3	Shipping prep.	Total throughput time	Order finish date
449	4	3	3	2	15	0	20	43	20-10-2021
443	4	2	2	2	14	0	9	29	20-10-2021
437	4	1	3	2	14	0	28	48	20-10-2021
433	4	2	4	2	17	0	16	41	20-10-2021
429	4	4	3	2	14	0	30	53	20-10-2021

Figure 14 Prinzen Dashboard

The dashboard consists out of four graphs and one table. Furthermore, the dashboard contains six buttons. The buttons are added because the user of the dashboard is able to simulate one day on the input sheet of the dashboard. The user must then be able to update the KPIs too, therefore each button updates a KPI.

The upper left graph shows the average time in days per phase over the orders finished in the past week. The button called “Update KPI Average time per phase” updates the graph. The upper right graph shows the number of orders in progress per phase. The button called “Update KPI Order in progress per phase” updates this KPI. The lower left graph shows the planned orders in progress per week. The colored lines represent the different phases after order releasing. The same lines can be found in the lower right graph which shows the number of FTEs per week. The button called “Update KPI Planned orders in progress” in between the graphs updates the left graph and the button called “Update KPI Number of FTEs per week” updates the right graph. The table at the bottom of the dashboard shows the last five orders that were not completed in time. The button called “Update KPI Orders with a too long throughput time” on the left updates this table, the right button called “Update KPI highlight departments with too long throughput time” updates the phases that did not complete this order in time. The cells light up if the order processing in that phase took too long.

In combination with the input sheet the team leader operations is able to simulate the order process of Prinzen. Although there is not much data available of the process with minimal input this dashboard is able to tell where problems occur. The macros used to derive the dashboard can be found in Appendix G: Macro’s.

7. Conclusion

In this chapter, the conclusion, evaluation, recommendations, and future research are provided. Each research question was related to a chapter, in the concluding section of these chapters the research question is answered. The conclusions support the answering of the main research question.

> *How can Prinzen get a clear image of what the current throughput time is concerning the planned throughput figures?*

This question is answered in section Conclusions7.1. After that the results are evaluated in 7.3. In section 7.2 the recommendations for Prinzen following this research are mentioned. The last section 7.4, will discuss future research.

7.1. Conclusions

The research aimed to give insights into the throughput time of Prinzen, such that the team leader operations is able to steer the process based on the information. To answer the main research question a method was designed based on literature. The method was redesigned and transformed to fit the situation of this specific company case. The designed method consists out of five steps, these steps are all taken to answer the main research question. The method can be found in chapter 5. The application of the method leads to a KPI dashboard made with dummy data, based on qualitative data gathered from interviews. The dashboard and KPIs are part of the quantitative research. That is why this research method can be described as a mixed research method.

The KPIs are made based on dummy data because the data necessary for these KPIs were not available. The lack of data made it difficult to do research. However, with the use of interviews representative dummy data was created. From this can also be concluded that the thought of the throughput time is too long at Prinzen is based on feelings from the practice rather than on available data. The most important data missing at Prinzen is an event log of the orders. Chapter 6.2.3 describes in detail what data measure points such an event log should at least contain. The order process flow or Prinzen can be divided into six phases, these can be found in Figure 2 The general process flow of Prinzen. The data measure points in the event log are based on the six phases of the order process flow. The throughput time is measured per phase such that the team leader operations is able to detect an issue per department.

Another aspect of measuring the throughput time is the desired throughput times. The orders are classified into three groups, 4, 8, or 12 weeks, according to the complexity and size of the order. A remarkable conclusion can be derived from qualitative research in the desired throughput time. The classification of the orders is not company-wide the same. For the image of the throughput time, this needs to be the same for every employee.

In conclusion, with the necessary data available the dashboard with its KPIs is able to give insight into the throughput time. Important is that the data must be available, however to show what is possible, dummy data was created. The user of the dashboard even has the possibility of simulating one day. Another important but necessary input for a clear image of the throughput time is to make sure that the classification of the orders is the same for every employee. If those two inputs are available the dashboard can give the problem owner, in this case, the team leader operations, a clear image of what the current throughput time is concerning the planned throughput figures, or in this case the KPIs.

7.2. Recommendations

This section describes the recommendations that could be useful for Prinzen to get an even better image of the throughput time. These recommendations are proposed based on the research.

Event log – To analyse the throughput time of each order an event log is necessary. The event log must not be too complex but also not too detailed. Figure 13 shows an example of a useful event log. The degree of detail in the event log results in a detailed analysis. I recommend Prinzen uses at least one measure date point for each phase, such that they are able to measure the throughput time for each phase. The event log should be very detailed, such that it gives more insight about not only what is happening but also why is this happening, i.e. what the real problems are that create delays.

Classification label – Early in the research it became clear that the classification of the order is not clear to all employees. A difference in this classification can affect the performance indicators a lot. Therefore a recommendation would be to add a label to each order with the classification (4, 8, or 12 weeks). This makes it easier to analyse the performance of certain orders. The classification is the desired throughput time, and therefore should be company-wide the same.

Data – Another recommendation is to gather data about the processing of orders. The input data for the dummy data is now based on qualitative answers from the interview. To use the simulation function of the dashboard as optimal as possible, the input data should be based on real data. Figure 12 shows the input of the dummy data, the averages, and the standard deviations mentioned there could be based on real data. If that is the case the simulation function will be of more value to Prinzen.

Accuracy – If the above recommendation is done, the simulation function must be checked on its accuracy. Does the simulated data represent the real data? The simulation is based on observations of employees of Prinzen, rather than based on the real date of the process. If the simulation will be based on real data, Prinzen must check the accuracy of the simulation model.

7.3. Evaluation

In this section, the limitations of the research will be discussed. These limitations could affect the validity of the research. The biggest limitation of this research was the lack of useful data. The data therefore was gathered in another way, namely by conducting interviews with representatives of the different departments of Prinzen. From this data, the dummy data was created. However, the data from the interviews are based on observations of the employees. These observations can be erroneous, also observations change from day to day. Every order has a different time, so the next day the same question about the throughput time can get a different answer. This is immediately the second limitation, the correctness of the data is a known problem. The data that is available can be falsified or not correct. The employees are looking at data that tells something else than the reality. According to the ERP system they have for instance multiple hundred hours backorder, but still manage to complete each order in time. The data then don't rhyme the reality. To answer the fourth research question the table in the dashboard which shows the latest five orders with a too long throughput time is able to detect the phase in which the issue will be. So the dashboard has a sufficient capability to allow for issues detection earlier during the process.

7.4. Future research

To even get a better image of the throughput time, some suggestions for further research are stated below. Some suggestions were already mentioned by employees in interviews but were out of the scope of this research.

Event log data – Future research can be done about the most efficient way of measuring the throughput time. Especially what data measure points should be added in the order process, and how those can be measured in the ERP system. It is known that Prinzen aims to have a new ERP system in 2023, so it immediately can be implemented in the new ERP system.

Live order tracking – This is the next-level way of order tracking via an event log. This was also mentioned by an employee of Prinzen. Searching for the place where an order is, but also the number of orders at a certain stage. With live order tracking, there are a lot more options in analyzing the order process flow. The team leader operations will then be able to act very fast if something is going wrong.

Value-added time – This KPI could yet not be implemented in the research. Therefore in future research, it would be nice to measure what the value-added time is per order. The value-added time also states something about the efficiency of the order process, especially the efficiency of each task performed on an order.

References

Abtew, W., Moras, R. G., & Campbell, K. (1990). Synthetic precipitation data generation. Computers & Industrial Engineering, 19(1–4), 582–586. https://doi.org/10.1016/0360-8352(90)90185-o

Barjis, J. (2008). The importance of business process modeling in software systems design. Science of Computer Programming, 71(1), 73–87. https://doi.org/10.1016/j.scico.2008.01.002

Blackburn, J. D. (ed.) (1991), Time-Based Competition: The Next Battleground in Industrial Competition, Business One Irwin, Homewood, IL

Cheng, H. J., & Kumar, A. (2015). Process mining on noisy logs — Can log sanitization help to improve performance? Decision Support Systems, 79, 138–149. https://doi.org/10.1016/j.dss.2015.08.003

Dietz, J.L.G., Barjis, J. & Goldkuhl, G. (2002) The atoms, molecules and matter of organizations. Proceedings of the Seventh International Workshop on the LAP, Delft, Netherlands.

Eckerson, W. W. (2006). Performance dashboards: Measuring, monitoring, and managing your business, Wiley, New York.

Ferreira, S., Silva, F., Casais, R., Pereira, M., & Ferreira, L. (2019). KPI development and obsolescence management in industrial maintenance. Procedia Manufacturing, 38, 1427–1435. https://doi.org/10.1016/j.promfg.2020.01.145

Golafshani, N. (2003). Understanding reliability and validity in qualitative research. Qualitative Report, 8, 597-606. Retrieved July 2, 2009 from http://www.nova.edu/ssss/QR/QR8-4/golafshani.pdf

Grigorova, K., Malysheva, E., & Bobrovskiy, S. (2017). Application of Data Mining and Process Mining approaches for improving e-Learning Processes. Data Science, 115–121. https://doi.org/10.18287/1613-0073-2017-1903-115-121

Heerkens, H., & Winden, A. (2021). Solving Managerial Problems Systematically (1ste editie). Wolters-Noordhoff.

Hruska, R., Prusa, P., & Babic, D. (2014). THE USE OF AHP METHOD FOR SELECTION OF SUPPLIER.

Transport, 29(2), 195-203. doi:10.3846/16484142.2014.930928

Huh, Y. U., Keller, F. R., Redman, T. C., & Watkins, A. R. (1990). Data quality. AT & T Bell laboratories, 32(8), 559–565.

Lederer, P. J., & Li, L. (1997). Pricing, Production, Scheduling, and Delivery-Time Competition. Operations Research, 45(3), 407–420. https://doi.org/10.1287/opre.45.3.407

Lindberg, C. F., Tan, S., Yan, J., & Starfelt, F. (2015). Key Performance Indicators Improve Industrial Performance. Energy Procedia, 75, 1785–1790. https://doi.org/10.1016/j.egypro.2015.07.474

Maskell, B., (1989) Performance measures of world class manufacturing, Management Accounting, pp. 32-3.

monday.com. (n.d.). Werk op de manier die voor jou werkt. Retrieved June 16, 2021, from https://monday.com/lang/nl/

Neely, A., Gregory, M., & Platts, K. (1995). Performance measurement system design. International Journal of Operations & Production Management, 15(4), 80–116. https://doi.org/10.1108/01443579510083622

Poel, V. I., & Royakkers, L. (2011). *Ethics, Technology, and Engineering: An Introduction* (1ste editie). Wiley-Blackwell.

Polit, D.F., & Beck, C.T. (2012). Nursing research: Generating and assessing evidence for nursing practice. (9th ed.). Philadelphia, PA: Wolters Kluwer. Richardson, L. (2000). Evaluating ethnography. Qualitative Inquiry, 6, 253-256

Rutberg, S., & Bouikidis, C. D. (2018). Focusing on the Fundamentals: A Simplistic Differentiation Between Qualitative and Quantitative Research. Nephrology Nursing Journal, 45(2), 209–213.

Sedrakyan, G., Leony, D., Muñoz-Merino, P. J., Kloos, C. D., & Verbert, K. (2017). Evaluating student-facing learning dashboards of affective states. In European Conference on Technology Enhanced Learning (pp. 224-237). Springer, Cham.

Sedrakyan, G., Mannens, E., & Verbert, K. (2019). Guiding the choice of learning dashboard visualizations: linking dashboard design and data visualization concepts. JOURNAL OF COMPUTER LANGUAGES, 50, 19-38.

Schmenner, R. W. (1988), “The Merit of Making Things Fast,” Sloan Management Review, 30, 1, 11–17

Shahin, A., & Mahbod, M. A. (2007). Prioritization of key performance indicators. International Journal of Productivity and Performance Management, 56(3), 226–240. https://doi.org/10.1108/17410400710731437

software oplossingen | Bedrijfsapplicaties. (n.d.). SAP. Retrieved June 16, 2021, from https://www.sap.com/netherlands/index.html

Stalk, G., Jr. (1988), “Time—The Next Source of Competitive Advantage,” Harvard Business Review, 66, 4, 41–51.

Stalk, G., Jr. & Hout T. M. (1990), Competing Against Time: How Time-Based Competition is Reshaping Global Markets, Free Press, New York.

Suk, S. J., Hwang, B. G., Dai, J., Caldas, C. H., & Mulva, S. P. (2012). Performance Dashboard for a Pharmaceutical Project Benchmarking Program. Journal of Construction Engineering and Management, 138(7), 864–876. https://doi.org/10.1061/(asce)co.1943-7862.0000503

Suri, R. (1994), “Common Misconceptions and Blunders in Implementing Quick Response Manufacturing,” Proceedings of the SME Autofact ’94 Conference, 23 pp.

Suri, R. AND de Treville, S. (1986), “Getting from ‘Just-in-Case’ to ‘Just-in-Time’: Insights from a Simple Model,” Journal of Operations Management, 6, 3, 295–304.

Tadanai, O., & Tanuska, P. (2015). Process mining and practical usage. 2015 IEEE 19th International Conference on Intelligent Engineering Systems (INES), 63–67. https://doi.org/10.1109/ines.2015.7329758

Tashakkori, A., & Creswell, J.W. (2007). The new era of mixed methods. Journal of Mixed Methods Research, 1(1), 3-7.

Tatsiopoulos, I. P., & Kingsman, B. G. (1983). Lead time management. European Journal of Operational Research 14, 351–358.

Tracy, S. J. (2010). Qualitative Quality: Eight “Big-Tent” Criteria for Excellent Qualitative Research. Qualitative Inquiry, 16(10), 837–851. https://doi.org/10.1177/1077800410383121

Velimirović, D., Velimirović, M., & Stanković, R. (2011). Role and importance of key performance indicators measurement. Serbian Journal of Management, 6(1), 63–72. https://doi.org/10.5937/sjm1101063v

Weick, K. E. (2007). The generative properties of richness. Academy of Management Journal, 50, 14-19.

Youngman, D. C. (2006). Process Flow Analysis in Academic Libraries. Technical Services Quarterly, 24(1), 37–44. https://doi.org/10.1300/j124v24n01_03

Appendices

Appendix A: Machines

Prinzen almost has 20 standardized machines. The machines can be put together to form complete handling lines. Not all machines are full-automatic, the Ovoset and the Ovoset Pro are semi-automatic. The machines are divided into three groups. One for the hatchery eggs, one for the consumption eggs, and the last for automation of the hatchery process.

Hatchery eggs:

Machine	Function	Capacity (eggs/hour)
Ovoset Pro	Point setting machine	30.000
Ovoset	Point setting machine	12.000
PSPC 5	Point setting machine	18.000
PSPC 7	Point setting machine	21.000
Trolley Loader	Load trays in trolley	30.000

Table 4 Hatchery eggs machines

Consumption eggs

Machine	Function	Capacity (eggs/hour)
Prinzen 50	Packing eggs	18.000
Prinzen 70	Packing eggs	25.200
Speedpack 110	Packing eggs	40.000
Speedpack 220	Packing eggs	80.000
Smartpack	Packing eggs	30.000
Smartpack twin	Packing eggs	60.000
Timeline	Pallet eggs	40.000
Easystack	Tray stacking	18.000
PS4 Traystacker	Tray stacking	40.000
Pallet loader	Pallet eggs	80.000

Table 5 Consumption eggs machines

Hatchery automation

Machine	Function	Capacity (eggs/hour)
Ovograder	Weighing and Sorting	30.000
Ovostamp	Inkjet print code on egg	-
Egg flex stamper	Stamping code on egg	-
Meggsius Family	Detect a leaking egg	-

Table 6 Hatchery automation machines

Appendix B: Conversation

Conversation Affairs & resource officer

The affairs & resource officer at Prinzen said three things were causing the difference between norm and reality. The first thing is the flexibility of the employees. Many of the employees are very good at making one machine because they did that a lot. If Prinzen introduces a new machine they cannot make the machine. Prinzen keeps track of the competencies of their employees in a competence matrix. It is difficult to find technologically skilled people. The second thing that causes the difference in throughput time is the bills of materials. Prinzen has 19 standard machines, but most of the time those machines need to change a little bit according to the wishes of the client. Then the employee of Prinzen manipulates the bill of

materials. This is a very risk-full thing to do. Therefore it happens often that a machine is not complete when it is finished. The last thing is the reliability of some of the suppliers. They deliver wrong or broken stuff and not in time. It is very difficult to tell where the long throughput time is exactly coming from. There are a lot of possible problems, but which problem is causing the long throughput time is difficult to say.

Conversation Teamleader Operations (COO)

Teamleader Operations at Prinzen gave me a bigger list of possible problems. The first thing he mentioned was also the non-flexibility of the employees. He plans the number of hours the employees should spend on a machine. Then because of the non-flexibility, the employees do not achieve that norm. It could also be said vice versa he is planning too tight. Another thing also mentioned by the affairs & resource officer was the bills of materials, but he told already that the company is working on that problem. The third thing that was also already mentioned by the affairs & resource officer, the reliability of the suppliers. The reliability of the suppliers can be divided into two things. First, they do not deliver on time, second, the materials that they deliver do not meet the requirements. Materials are for instance damaged. The next thing he told that might cause the difference was when there is a gap in production, because of suppliers delay, employees ask if they can go home because they cannot work. There is work, but they are not qualified enough to do that job. This comes back to the flexibility of the employees. It happens sometimes that the warehouse workers pick the wrong items. Then the item must return to the warehouse, the COO told that this cost a lot of time. Sometimes there are also mistakes with the layout of the bills of materials. Clients want to change or expand their products during the production process. He also told me to ask projects how long they wait with entering the sales order. He thought they do this every two days (Monday, Wednesday, Friday). This can also become a problem. The last thing that the COO told was about recognizing problems. Problems often are unrecognized until fine-tuning of the machines. They do this with a so-called crack test. The results of the crack test are also not very clear. When has a machine passed the crack test? If they find a problem in fine-tuning, they have to break down the machine and build it up again.

Conversation Area Sales Manager

The Area Sales Manager is responsible for clients in the eastern countries of Europe. He has direct contact with the dealers of Prinzen. His biggest problem is that Prinzen is not flexible anymore. He worked in production a couple of years back. Back then, it was more focused on being flexible. Machines could be built in less than three weeks. Clients could change their machines according to their preferences. Production workers switched a lot between machines. For a couple of years, this is not the case anymore. Production workers are working on a machine and will not change from the machine until it is finished. Also, the stock is very small, almost every material in stock is already matched with a machine. Therefore Prinzen is not flexible anymore. He would like it when Prinzen gets more flexible. Luckily in his countries, clients do not have a lot of wishes, but he knows that the sales manager of Germany has many clients that wish to change their product a bit. Therefore there is a lot of discussions in the company between production and the sales manager of Germany. This is because production does not want to change a lot on the machines, it also causes a lot of troubles in production.

Conversation Team Leader Sales Support

The team leader sales support told me more about the cause of the problems that occur in Prinzen today. He told that in the last years Prinzen grew a lot. Prinzen was focused on selling opportunities. They did not 'finish' everything very well, for instance bills of materials were not complete. During the growing phase that did not matter, but now the growth is stagnating. He notices that it does a lot of 'pain'. Prinzen is working hard on finishing all the machines and processes. Another big problem he stated is that the new generations are used to fast delivery times. People are living in a 24/7 buying society. The generation that buys the machines also expects a very short delivery time. The generation that sells the machines is not used to very short delivery times. This leads to difficult negotiations. Price does not matter a lot to those buyers anymore. If you cannot deliver fast you do not even have to send an offer. That is why delivery times and thus throughput times are very important.

Conversation Plant manager (CEO)

The CEO started in 2017 at Prinzen. He has changed a lot since then. Prinzen did a lot of specials. Specials are machines that are not standard. Almost every customer had his wishes. The management of Prinzen was very hierarchical. That is why he changed a lot. Prinzen came up with three strives of the company:

- Self-direction: employees of Prinzen should start when they want to, they should learn when they want to. It does not matter which employee is working on which machine. The employees can decide all of that by themselves.
- Configure to order: All machines should be configured to order. This is because Prinzen wants its innovation department to be busy with innovations, not with the engineering wishes of the clients on already existing machines.
- Throughput times of 4, 8, or 12 weeks: to meet customer expectations Prinzen wants throughput times of 4, 8, or 12 weeks. This is because the new generation of customers is asking for fast delivery times.

The last strive is the most important for the research. What is the thought behind the 4, 8, or 12 weeks? The CEO told me that the workload is less long in the company. This is better because more workload means more space, more time, and more distraction. Employees are working on a machine even when it is just standing in the company, they still need to check on the machine once and then. The 4, 8, or 12 weeks are also because it is a mindset. It is easy to remember for everybody, and everybody knows that it is the goal. So all employees at Prinzen need to work together on reaching that goal.

The throughput time starts from the moment the customer confirms the order and ends when the machine leaves Prinzen. What machines have which target throughput time? One functional machine has a target time of four weeks. When more functional machines are combined the target time is 8 weeks. And when Prinzen needs to build complete lines the target time is 12 weeks. From this, he concluded that it is better to start with just one functional machine. And then give insights into the throughput time of those orders. He also told me something about eggficiency. This is a project mapping the process inside Prinzen. More information about eggficiency can be found in appendix D.

Appendix C: Order type

Figure 15 Order type

Appendix D: Eggfiecency

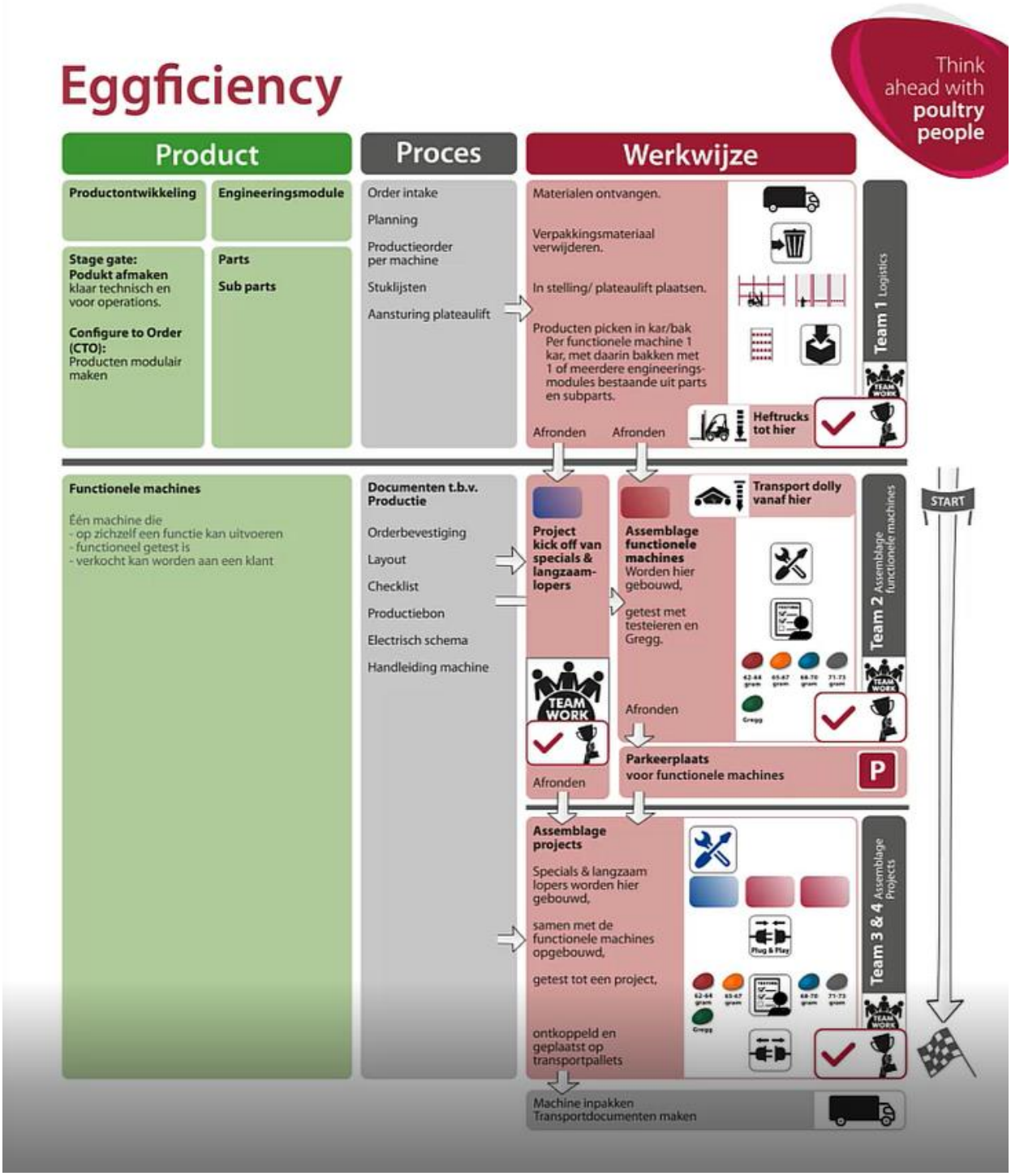


Figure 16 Eggficiency poster

In the poster above the eggficiency plan of Prinzen is described. The poster describes the process from above to below. The product can be found in the left column, the middle column describes what process is taking place, and the right column describes how the process should be done. The process starts with order intake, the order can be standard or the order needs engineering. The order will go to work planning or engineering based on the type of order. After engineering, the order will also go to work planning. Work planning plans the order and the order will go to logistics, the order will be picked out of the storage and placed on a shelf on

wheels. One machine can be built from one shelf on wheels, all parts are present on the shelf. Then the shelf on wheels goes to production. Production assembles the order, fine-tune the order and then the order is finished. Then the order is packed for delivery, and shipped by the delivery company.

What is so eggfiecent in this desired process? In every process, they should strive for efficiency. In logistics, all packing materials should be removed from the parts. The forklift truck should never be on the production floor. Only order pickers can pick parts from storage. In production the shelves are on wheels, this is because the production worker has then a flexible workspace. Every squared meter on the production floor can be used to build a machine. So every process has its efficiency desires.

Appendix E: Interview

1. Welke functie heb je bij Prinzen? Wat voor werk doe je dagelijks?

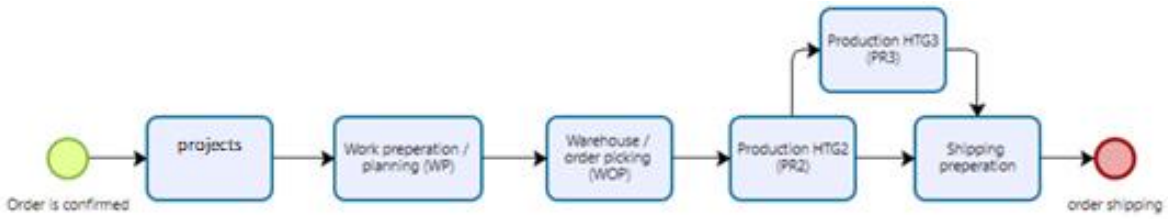
Functie:	Werk:

2. Kun je het onderstaande schema invullen gebaseerd op het eggfiecency plan? (zet een X in de juiste kolom)

Machine	4 weken	8 weken	12 weken
Ovoset pro			
Ovoset			
PSPC 5			
PSPC 7			
Trolley Loader			
Prinzen 50			
Prinzen 70			
Speedpack 110			
Speedpack 220			
Smartpack			
Smartpack twin			
Timeline			
Easystack			
PS4 Traystacker			
Pallet loader			
Ovograder			
Ovostamp			
Egg flex stamper			
Meggsius Family			

3. Kun je de onderstaande tabel invullen op basis van de orders op de volgende pagina's? Vul eerst in hoe lang de doorlooptijd van een order mag zijn, door een kruisje te zetten in de juiste kolom. Vul daarna in hoelang jij bezig bent met de order in de laatste kolom.

Order	4 weken	8 weken	12 weken	Tijd mee bezig
A				
B				
C				
D				
E				
F				
G				
H				



Als we het process van een order in 6 fases verdelen, 1. Projects, 2. Werk voorbereiding, 3. Magazijn, 4. Productie htg2, 5. Productie htg3, 6. Inpakken.

4. In welke fase werk jij? (Indien je niet in een van deze fases werkt vul hier dan niets in, zit je tussen een van de fases zet dan een kruisje op die plek hierboven in de afbeelding.)

5. Welke informatie/product(en) krijg je van de vorige fase?

6. Wat doe jij met deze informatie/product? (beschrijf dit zo gedetailleerd mogelijk, stap voor stap)

7. Wat geef jij aan informatie/product(en) door aan de volgende fase?

8. Hoe lang mag een order bij de verschillende fases zijn? (in percentage, of in het geval van een order van 4 weken)

Fase	Order 4 weken (tijd)	Percentage voor alle orders
1. Sales		
2. Werk voorbereiding		
3. Magazijn		
4. HTG 2		
5. HTG 3		
6. Logistiek		

9. Welke gegevens zou jij willen hebben met betrekking tot de doorlooptijd? Bijvoorbeeld tijd over het hele process, of tijd per fase. Afgezet tegen de doeltijd. (KPI's)

10. Op een schaal van 1 tot 10, hoe belangrijk is het dat deze gegevens makkelijk te achterhalen zijn in een ERP pakket zoals SAP.

a.

Pos.	Artikel	Orderhoeveelheid	HE	I	Omschrijving	E
1				☐	documenten	
10	9108000132	1	ST	☐	Prinzen 50	
20	9600100192	1	ST	☐	Prinzen 50 basic	
30	9600100195		ST	☐	Prinzen 50 Version RB	
40	9600100198	1	ST	☐	Extended infeed belt 550*605	
50	9600100022	1	ST	☐	Vacuüm transfer 17lbs tray	
60	9600100125	1	ST	☐	Denesterset paper for PSPC/P50	
70	9600100202	1	ST	☑	Ergostack 50	

b.

Alle posities

Pos.	Artikel	Orderhoeveelheid	HE	I	Omschrijving	Be
1				☐	documenten	
11				☐	Machine 1:	
21	9108000009	1	ST	☑	PSPC 5 Automatischer Eierpacker	
31	9104700120	1	ST	☑	Int. Verlängerung L550mm PSPC..	
41	9104000032	1	ST	☐	Ovograder 20	
51	9600100238	1	ST	☐	Ovograder 20 Basic	
61	9600100240		ST	☐	Ovograder besturingskast links	
71	9104700029	1	ST	☑	Dosiertisch L=1400xB=710mm	
81	9104700090	1	ST	☑	Satz Ersatzteile für PSPC&Ovoset	
91	9600100244	1	ST	☑	Set of spares for Ovograder 20	
101	9608300001	1	ST	☑	System Configuratie	
111				☐	Machine 2:	
121	9108000009	1	ST	☑	PSPC 5 Automatischer Eierpacker	
131	9104700120	1	ST	☑	Int. Verlängerung L550mm PSPC..	
141	9104000032	1	ST	☐	Ovograder 20	
151	9600100238	1	ST	☐	Ovograder 20 Basic	
61	9600100240		ST	☐	Ovograder besturingskast links	
71	9104700029	1	ST	☑	Dosiertisch L=1400xB=710mm	
81	9104700090	1	ST	☑	Satz Ersatzteile für PSPC&Ovoset	
91	9600100244	1	ST	☑	Set of spares for Ovograder 20	
101	9608300001	1	ST	☑	System Configuratie	
111				☐	Machine 2:	
121	9108000009	1	ST	☑	PSPC 5 Automatischer Eierpacker	
131	9104700120	1	ST	☑	Int. Verlängerung L550mm PSPC..	
141	9104000032	1	ST	☐	Ovograder 20	
151	9600100238	1	ST	☐	Ovograder 20 Basic	
161	9600100240		ST	☐	Ovograder besturingskast links	
171	9104700029	1	ST	☑	Dosiertisch L=1400xB=710mm	
181	9104700090	1	ST	☑	Satz Ersatzteile für PSPC&Ovoset	
191	9600100244	1	ST	☑	Set of spares for Ovograder 20	
201	9608300001	1	ST	☑	System Configuratie	
211	1130200001	1	ST	☐	Transportkosten	
221				☐	Payment conditions:	
231	DOWN PAYMENT.	1	ST	☐	50% Anzahlung	
241	DOWN PAYMENT.	1	ST	☐	50% vor Lieferung (spatestens …	

c.

Alle posities

Pos.	Artikel	Orderhoeveelheid	HE	I	Omschrijving
1				☐	documenten
11	9108000132	1	ST	☐	Prinzen 50
21	9600100192	1	ST	☐	Prinzen 50 basique
31	9600100196		ST	☐	Prinzen 50 Version LB
41	9600100197	1	ST	☐	Bande d'alimentation standard
51	9600100022	1	ST	☐	Vacuüm transfer 17lbs tray
61	9600100125	1	ST	☐	Denesterset paper for PSPC/P50
66	9600100126	1	ST	☐	Denester set plastic for PSPC/P50
71	9600100201	1	ST	☑	Ovostamp 5R PR50, 5 tampons+...
81	9600100202	1	ST	☑	Ergostack 50
91	9104700006	1	ST	☑	Doseur L=1150mm x L=605mm
101	1130200001	1	ST	☐	Prix de Transport
111	3000304352	4	ST	☐	Vérin Easystack

d.

Alle posities

Pos.	Artikel	Orderhoeveelheid	HE	I	Omschrijving
1				☐	documenten
11	9108000009	1	ST	☑	Prinzen emballeuse type PSPC 5
21	9104700004	1	ST	☑	Rallonge intégral L=550mm x L=...
31	9104700090	1	ST	☑	Jeu de pièces de rechange PSPC...
41	1130200001	1	ST	☐	Prix de Transport

e.

10	7100200037	4	ST	☐	TR1-3026 Plastic sprocket infeed...
20	6900100031	30	ST	☐	Foodgrade ink black 125 ml.
30	8500400021	3	ST	☐	Drive roll 37:1 50x1,5 24vdc
40	6400400006	3	ST	☐	Vacuumfilter VTF01-118
50	6400700012	2	ST	☐	Coil
60	8001000001	3	ST	☐	Optocoupler 24DC RP1D0606D8...
70	6400300008	1	ST	☐	Valve, rod Ni. incl. cables
90	8001000162	2	ST	☐	Relay TM3SAC5RG
100	6200700013	4	ST	☐	Valve GRLA-1/8-QS-6-D
110	4000200073	100	ST	☐	Cup white plastic
120	6900200005	6	ST	☐	STG Cartridge holder channel 1
130	6900200008	1	ST	☐	Main cable 5 head 100cm
140	6100200272	2	ST	☐	Conveyorbelt drive roller - 5R
150	7200100021	2	ST	☐	Conveyor belt 4320x58+PVU-8T...
160	6900100035	200	ST	☐	Cartridge red 959 N31 food-grad...
161				☐	With stock order
170				☐	
171				☐	Expected availability Week 29
180	8500400013	1	ST	☐	Motor RGM 54W 200rpm

f.

1				☐	documenten
10				☐	Machine 1:
20	9108000131	1	ST	☐	Prinzen 70
30	9600100015	1	ST	☐	Prinzen 70 basic
40	9600100016		ST	☐	Prinzen 70 Version R
50	9600100069	1	ST	☐	Extension L=550mm P70-Ovogr...
60	9600100022	1	ST	☐	Vacuüm transfer 17lbs tray
70	9600100024	1	ST	☐	Exchangeset gripperdenester un...
80	9104000033	1	ST	☐	Ovograder 30
90	9600100239	1	ST	☐	Ovograder 30 basic
100	9600100240		ST	☐	Ovograder panel left
110	9600100242	1	ST	☐	Set packing table
120	9104300016	1	ST	☑	Tray stacker PS4 for Prinzen 70 ...
130	9600100052	6	ST	☑	Gripper PS-4 quick change #2
140	9108700037	1	ST	☑	Conveyor set manual egg lifter
150	9600100041	1	ST	☑	Flattop conveyor L=2m
160	9104700069	1	ST	☑	Inspection rollerconveyor L=142...
170	9600100245	1	ST	☑	Set of spares for Ovograder 30
180	9104700183	1	ST	☑	Set of spares for PS4
190	9104700200	1	ST	☑	Set of spares for prinzen 70
200	1800200003	2	DAG	☐	Supervisor Prinzen per day
210	1800200016	1	ST	☐	Travel expenses incl. ticket
220	9608300001	1	ST	☑	System Configuration
230				☐	Machine 2:
240	9108000131	1	ST	☐	Prinzen 70
250	9600100015	1	ST	☐	Prinzen 70 basic
260	9600100016		ST	☐	Prinzen 70 Version R
270	9600100069	1	ST	☐	Extension L=550mm P70-Ovogr...
280	9600100022	1	ST	☐	Vacuüm transfer 17lbs tray
290	9600100024	1	ST	☐	Exchangeset gripperdenester un...
300	9104000033	1	ST	☐	Ovograder 30
310	9600100239	1	ST	☐	Ovograder 30 basic
320	9600100240		ST	☐	Ovograder panel left
330	9600100242	1	ST	☐	Set packing table
340	9104300016	1	ST	☑	Tray stacker PS4 for Prinzen 70 ...
350	9600100052	6	ST	☑	Gripper PS-4 quick change #2
360	9108700037	1	ST	☑	Conveyor set manual egg lifter
370	9600100041	1	ST	☑	Flattop conveyor L=2m
380	9104700069	1	ST	☑	Inspection rollerconveyor L=142...
390				☐	Machine 3:
400	9108000132	1	ST	☐	Prinzen 50
410	9600100192	1	ST	☐	Prinzen 50 basic
420	9600100193		ST	☐	Prinzen 50 Version R
430	9600100197	1	ST	☐	Standard infeed belt
440	9600100022	1	ST	☐	Vacuüm transfer 17lbs tray
450	9600100126	1	ST	☐	Denesterset plastic for PSPC/P50
460	8200300156	1	ST	☐	E-Stop Kit X68/X69/X600
470	9600100202	1	ST	☑	Ergostack 50
480	9104700069	1	ST	☑	Inspection rollerconveyor L=142...
490	9600100203	1	ST	☑	Set of spares for Prinzen 50
500	9600100205	1	ST	☑	Set of spares for Ergostack 50
510	1800200003	1	DAG	☐	Supervisor Prinzen per day
520	1800200016	1	ST	☐	Travel expenses incl. ticket
530	9600100209	1	ST	☑	System configuration
540	1130200001	1	ST	☐	Transport Costs

g.

1				documenten
11	9108000132	1	ST	Prinzen 50
21	9600100192	1	ST	Prinzen 50 basic
31	9600100196		ST	Prinzen 50 Uitvoering LB
41	9600100197	1	ST	Standaard invoermat
51	9600100232	1	ST	Standaard uitvoer P50
61	9600100022	1	ST	Vacuüm transfer 17lbs tray
71	9104700008	1	ST	Dosiertisch L=1150mm x B=113...
81	1130200001	1	ST	Transportkosten

h.

1				Documenten
10	9108000134	1	ST	Ovoset for 30 JW42- 84 CM165-...
20	9108800003	1	ST	Toolkit hatchery packer
30	9600100206	1	ST	Meggsius Detect
40	9104700206	1	ST	Set of spares Ovoset
50	9508100040	3	ST	Egg guide set VB40
60	9508100034	9	ST	Leg set with adjustable foot galv..
70	9508100036	26	ST	Chain, carriers and sliding profile ...
80	9508100022	4	ST	Section VB40
90	9508100032	2	ST	Decline set VB40
100	9508100030	2	ST	Incline set VB40
110	9508100002	1	ST	Drive unit VB40 right
120	9500100223	1	ST	Right-angle motorset VB drive 4...
130	9508100014	1	ST	Return unit VB40
140	9008100089	4	ST	Egg transition from US nest to V...
150	9508100074	1	ST	Tool set VB40

Appendix F: Machine classification interview

Machines	Affairs & resource officer	Operations planning officer	Process engineer	Work preparation	Project leader	Project leader
1. Ovoset pro	4	8	8	8	4	8
2. Ovoset	4	4	4	4	4	8
3. PSPC 5	4	4	4	4	4	8
4. PSPC 7	4	8	8	4	4	8
5. Trolley Loader	8	8	8	12	8	8
6. Prinzen 50	4	4	4	4	4	4
7. Prinzen 70	4	4	4	4	4	4
8. Speedpack 110	4	4	8	12	4	8
9. Speedpack 220	8	8	12	12	8	8
10. Smartpack	4	4	8	8	4	8

11. Smartpack twin	8	8	8	12	8	8
12. Easystack	4	4	4	4	4	4
13. PS4 Traystacker	4	4	4	4	4	4
14. Pallet loader	12	12	12	12	8	12
15. Ovograder	4	4	4	4	4	4
16. Ovostamp	4	4	4	4	4	4

Table 7 interview results

Machines	Managing director	Team leader operations
1. Ovoset pro	8	8
2. Ovoset	4	4
3. PSPC 5	4	4
4. PSPC 7	4	8
5. Trolley Loader	8	8
6. Prinzen 50	4	4
7. Prinzen 70	4	4
8. Speedpack 110	4	4
9. Speedpack 220	8	4
10. Smartpack	4	4
11. Smartpack twin	8	8
12. Easystack	4	4
13. PS4 Traystacker	4	8
14. Pallet loader	12	12
15. Ovograder	4	4
16. Ovostamp	4	4

Table 8 Interview results

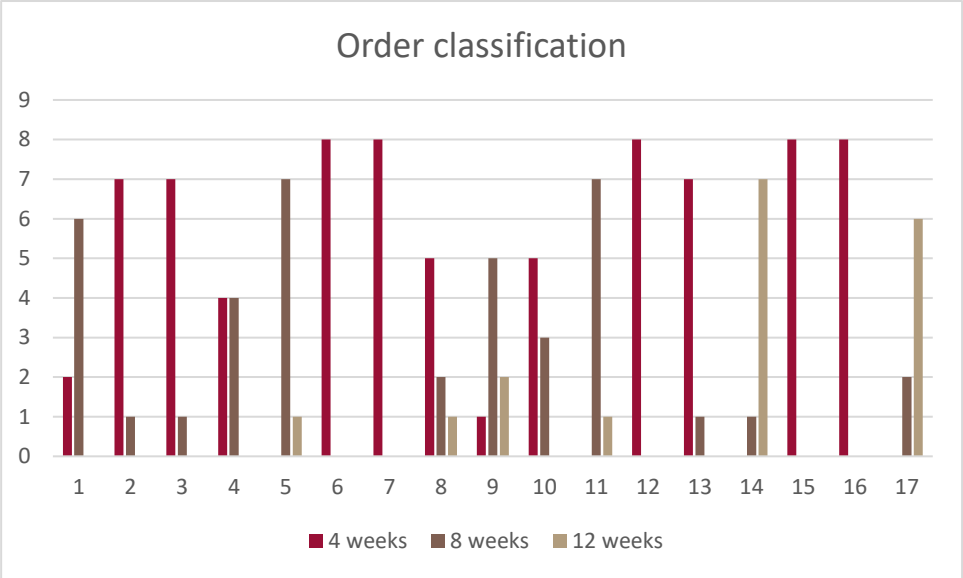


Figure 17 order classification

Appendix G: Macro's

Appendix G.1: macro generating event log data

```
Sub Eventlogcreater()

Dim sh1 As Worksheet
Dim sh2 As Worksheet

Set sh1 = ActiveWorkbook.Sheets("Input dummy data")
Set sh2 = ActiveWorkbook.Sheets("Eventlog")

'all variables are created
Dim i, j, k As Long
Dim startdate As Long
Dim today As Long
Dim Ordernr As Long
Dim AvgNumOrdDay, DevNumOrdDay, MinNumOrdDay, MaxNumOrdDay As Double
Dim AvgProTime, DevProTime, MinProTime, MaxProTime, NormProTime As Double
Dim AvgWorkPrepTime, DevWorkPrepTime, MinWorkPrepTime, MaxWorkPrepTime, NormWorkPrepTime As Double
Dim AvgWarehouseTime, DevWarehouseTime, MinWarehouseTime, MaxWarehouseTime, NormWarehouseTime As Double
Dim AvgHTG2Time4, DevHTG2Time4, MinHTG2Time4, MaxHTG2Time4, NormHTG2Time4 As Double
Dim AvgHTG2Time8, DevHTG2Time8, MinHTG2Time8, MaxHTG2Time8, NormHTG2Time8 As Double
Dim AvgHTG2Time12, DevHTG2Time12, MinHTG2Time12, MaxHTG2Time12, NormHTG2Time12 As Double
Dim AvgHTG3Time8, DevHTG3Time8, MinHTG3Time8, MaxHTG3Time8, NormHTG3Time8 As Double
Dim AvgHTG3Time12, DevHTG3Time12, MinHTG3Time12, MaxHTG3Time12, NormHTG3Time12 As Double
Dim AvgOrderFinishTime, DevOrderFinishTime, MinOrderFinishTime, MaxOrderFinishTime, NormOrderFinishTime As Double


Dim dblNumberOrders As Double
Dim intNumberOrders As Long
Dim OrderClass As Integer
Dim rndOrderClass As Double
Dim rndWishedDate As Integer
Dim Weekday As Integer
Dim WeekdayWishedDate As Long
Dim ProjectsTime As Long
Dim EndProjectsDate As Long
Dim WorkprepTime As Long
Dim EndWorkprepDate As Long
Dim EndwarehouseDate As Long
Dim EndHTG2Date As Long
Dim EndHTG3Date As Long
Dim OrderFinishDate As Long

Dim OrderReleaseDate1, OrderReleaseDate2, OrderReleaseDate3 As Long
Dim WeekdayORD1, WeekdayORD2, WeekdayORD3 As Long
Dim rand1, rand2, rand3, rand4, rand5 As Long


'get variables input from input sheet
startdate = sh1.Cells(2, 4)
today = sh1.Cells(3, 4)
AvgNumOrdDay = sh1.Cells(9, 3)
DevNumOrdDay = sh1.Cells(9, 4)
MinNumOrdDay = sh1.Cells(9, 5)
```

```
MaxNumOrdDay = sh1.Cells(9, 6)
AvgProTime = sh1.Cells(10, 3)
DevProTime = sh1.Cells(10, 4)
MinProTime = sh1.Cells(10, 5)
MaxProTime = sh1.Cells(10, 6)
AvgWorkPrepTime = sh1.Cells(11, 3)
DevWorkPrepTime = sh1.Cells(11, 4)
MinWorkPrepTime = sh1.Cells(11, 5)
MaxWorkPrepTime = sh1.Cells(11, 6)
AvgWarehouseTime = sh1.Cells(12, 3)
DevWarehouseTime = sh1.Cells(12, 4)
MinWarehouseTime = sh1.Cells(12, 5)
MaxWarehouseTime = sh1.Cells(12, 6)
AvgHTG2Time4 = sh1.Cells(13, 3)
DevHTG2Time4 = sh1.Cells(13, 4)
MinHTG2Time4 = sh1.Cells(13, 5)
MaxHTG2Time4 = sh1.Cells(13, 6)
AvgHTG2Time8 = sh1.Cells(14, 3)
DevHTG2Time8 = sh1.Cells(14, 4)
MinHTG2Time8 = sh1.Cells(14, 5)
MaxHTG2Time8 = sh1.Cells(14, 6)
AvgHTG2Time12 = sh1.Cells(15, 3)
DevHTG2Time12 = sh1.Cells(15, 4)
MinHTG2Time12 = sh1.Cells(15, 5)
MaxHTG2Time12 = sh1.Cells(15, 6)
AvgHTG3Time8 = sh1.Cells(16, 3)
DevHTG3Time8 = sh1.Cells(16, 4)
MinHTG3Time8 = sh1.Cells(16, 5)
MaxHTG3Time8 = sh1.Cells(16, 6)
AvgHTG3Time12 = sh1.Cells(17, 3)
DevHTG3Time12 = sh1.Cells(17, 4)
MinHTG3Time12 = sh1.Cells(17, 5)
MaxHTG3Time12 = sh1.Cells(17, 6)
AvgOrderFinishTime = sh1.Cells(18, 3)
DevOrderFinishTime = sh1.Cells(18, 4)
MinOrderFinishTime = sh1.Cells(18, 5)
MaxOrderFinishTime = sh1.Cells(18, 6)

j = 1

'clears sheet two
sh2.Cells.Clear
sh2.Cells(1, 1) = "OrderNr"
sh2.Cells(1, 2) = "OrderClass"
sh2.Cells(1, 3) = "WishedDeliveryDate"
sh2.Cells(1, 4) = "OrderEntryDate"
sh2.Cells(1, 5) = "EndProjectsDate"
sh2.Cells(1, 6) = "EndWorkprepDate"
sh2.Cells(1, 7) = "OrderReleaseDate"
sh2.Cells(1, 8) = "EndWarehouseDate"
sh2.Cells(1, 9) = "EndHTG2Date"
sh2.Cells(1, 10) = "EndHTG3Date"
sh2.Cells(1, 11) = "OrderFinishDate"
sh2.Cells(1, 12) = "PlannedOrderReleaseDate"
For i = startdate To today 'loop over all past days
    Weekday = WorksheetFunction.Weekday(CDate(i))
    If Weekday < 7 And Weekday > 1 Then 'orders are not put into the system in the weekends
        dblNumberOrders = WorksheetFunction.Norm_Inv(Rnd(), AvgNumOrdDay, DevNumOrdDay)
        'with a normal distribution the number of orders per day is generated
        If dblNumberOrders < MinNumOrdDay Then
            dblNumberOrders = MinNumOrdDay
        ElseIf dblNumberOrders > MaxNumOrdDay Then
            dblNumberOrders = MaxNumOrdDay
        End If

        intNumberOrders = Int(dblNumberOrders)
        If intNumberOrders < 1 Then 'the least amount of orders per day is one
        intNumberOrders = 1
        End If

        For k = 1 To intNumberOrders
            sh2.Cells(j + 1, 1) = "400" + j 'ordernumber creation

            rndOrderClass = Rnd() 'determine order class with rates 60/30/10
        If rndOrderClass < 0.6 Then
            OrderClass = 4
        ElseIf rndOrderClass < 0.9 Then
            OrderClass = 8
        Else
            OrderClass = 12
        End If
        sh2.Cells(j + 1, 2) = OrderClass
        sh2.Cells(j + 1, 4) = CDate(i) 'put in order entry date
        rand1 = Rnd()
        If Int(WorksheetFunction.Norm_Inv(rand1, 5, 3)) < 0 Then
        'generate a wished delivery date for each order
        'the average is five weeks + the order class according to the teamleader operations
            rndWishedDate = 1
        Else
            rndWishedDate = Int(WorksheetFunction.Norm_Inv(rand1, 5, 3)) + 1
        End If
        rand2 = Rnd() 'Prinzen does not deliver in the weekend
        If WorksheetFunction.Weekday(CDate(i + (rndWishedDate + OrderClass - 1) * 7 + Int(rand2 * 7 + 1))) = 7 Then
            sh2.Cells(j + 1, 3) = CDate(i + (rndWishedDate + OrderClass - 1) * 7 + Int(rand2 * 7))
        ElseIf WorksheetFunction.Weekday(CDate(i + (rndWishedDate + OrderClass - 1) * 7 + Int(rand2 * 7 + 1))) = 1 Then
            sh2.Cells(j + 1, 3) = CDate(i + (rndWishedDate + OrderClass - 1) * 7 + Int(rand2 * 7 - 1))
        Else
            sh2.Cells(j + 1, 3) = CDate(i + (rndWishedDate + OrderClass - 1) * 7 + Int(rand2 * 7 + 1))
        End If

        'each time a phase time is generated this is done with a normal distribution, the time is added to the previous date.
        'Then again is checked wheter this new date is in the weekend or not. Then the data is put into the sheet.

        NormProTime = WorksheetFunction.Norm_Inv(Rnd(), AvgProTime, DevProTime) 'projects time is generated
        If NormProTime < MinProTime Then
            NormProTime = MinProTime
```

```
ElseIf NormProTime > MaxProTime Then
    NormProTime = MaxProTime
End If
ProjectsTime = Int(NormProTime * OrderClass / 4)
If ProjectsTime < 1 Then
    EndProjectsDate = i + 1
Else
    EndProjectsDate = i + ProjectsTime
End If
If WorksheetFunction.Weekday(CDate(EndProjectsDate)) = 1 Then
    EndProjectsDate = EndProjectsDate + 1
ElseIf WorksheetFunction.Weekday(CDate(EndProjectsDate)) = 7 Then
    EndProjectsDate = EndProjectsDate + 2
End If

If EndProjectsDate < today Then
    sh2.Cells(j + 1, 5) = CDate(EndProjectsDate)
End If
 'work preperation time is generated
NormWorkPrepTime = WorksheetFunction.Norm_Inv(Rnd(), AvgWorkPrepTime, DevWorkPrepTime)
If NormWorkPrepTime < MinWorkPrepTime Then
NormWorkPrepTime = MinWorkPrepTime
ElseIf NormWorkPrepTime > MaxWorkPrepTime Then
NormWorkPrepTime = MaxWorkPrepTime
End If
WorkprepTime = Int(NormWorkPrepTime * OrderClass / 4)
If WorkprepTime < 1 Then
    EndWorkprepDate = EndProjectsDate + 1
Else
    EndWorkprepDate = EndProjectsDate + WorkprepTime
End If

If WorksheetFunction.Weekday(CDate(EndWorkprepDate)) = 1 Then
    EndWorkprepDate = EndWorkprepDate + 1
ElseIf WorksheetFunction.Weekday(CDate(EndWorkprepDate)) = 7 Then
    EndWorkprepDate = EndWorkprepDate + 2
End If
If EndWorkprepDate < today Then
    sh2.Cells(j + 1, 6) = CDate(EndWorkprepDate)
    OrderReleaseDate1 = CLng(sh2.Cells(j + 1, 3)) - 3                    'three days for packing
WeekdayORD1 = WorksheetFunction.Weekday(CDate(OrderReleaseDate1))
If WeekdayORD1 = 1 Or WeekdayORD1 = 7 Or WeekdayORD1 = 6 Then
    OrderReleaseDate1 = OrderReleaseDate1 - 2
End If

If OrderClass = 4 Then                                              'number of days for assembly
    OrderReleaseDate2 = OrderReleaseDate1 - 14
ElseIf OrderClass = 8 Then
    OrderReleaseDate2 = OrderReleaseDate1 - 28
ElseIf OrderClass = 12 Then
    OrderReleaseDate2 = OrderReleaseDate1 - 56
End If

OrderReleaseDate3 = OrderReleaseDate2 - 2                           'two days for order picking
WeekdayORD3 = WorksheetFunction.Weekday(CDate(OrderReleaseDate2))
If WeekdayORD3 = 1 Or WeekdayORD3 = 7 Then
    OrderReleaseDate3 = OrderReleaseDate3 - 2
End If
sh2.Cells(j + 1, 12) = CDate(OrderReleaseDate3)
End If
'order release date is generated this is done according to information from the teamleader operations.
'Prinzen plans backward from the wished delivery date.

If CLng(sh2.Cells(j + 1, 12)) <= today And CLng(sh2.Cells(j + 1, 12)) > 0 Then
    sh2.Cells(j + 1, 7) = CDate(sh2.Cells(j + 1, 12))
End If

NormWarehouseTime = WorksheetFunction.Norm_Inv(Rnd(), AvgWarehouseTime, DevWarehouseTime) 'warehouse time
If NormWarehouseTime < MinWarehouseTime Then
    NormWarehouseTime = MinWarehouseTime
ElseIf NormWarehouseTime > MaxWarehouseTime Then
    NormWarehouseTime = MaxWarehouseTime
End If

If NormWarehouseTime < 1 Then
    EndwarehouseDate = OrderReleaseDate3 + 1
Else
    EndwarehouseDate = OrderReleaseDate3 + Int(NormWarehouseTime)
End If

If WorksheetFunction.Weekday(CDate(EndwarehouseDate)) = 1 Then
    EndwarehouseDate = EndwarehouseDate + 1
ElseIf WorksheetFunction.Weekday(CDate(EndwarehouseDate)) = 7 Then
    EndwarehouseDate = EndwarehouseDate + 2
End If

If EndwarehouseDate < today Then
    sh2.Cells(j + 1, 8) = CDate(EndwarehouseDate)
End If

If OrderClass = 4 Then                                                                  'production HTG2 time
    NormHTG2Time4 = WorksheetFunction.Norm_Inv(Rnd(), AvgHTG2Time4, DevHTG2Time4)
    If NormHTG2Time4 < MinHTG2Time4 Then
        NormHTG2Time4 = MinHTG2Time4
    ElseIf NormHTG2Time4 > MaxHTG2Time4 Then
        NormHTG2Time4 = MaxHTG2Time4
    End If
    If NormHTG2Time4 < 1 Then
        EndHTG2Date = EndwarehouseDate + 1
    Else
        EndHTG2Date = EndwarehouseDate + Int(NormHTG2Time4)
    End If
ElseIf OrderClass = 8 Then
    NormHTG2Time8 = WorksheetFunction.Norm_Inv(Rnd(), AvgHTG2Time8, DevHTG2Time8)
    If NormHTG2Time8 < MinHTG2Time8 Then
        NormHTG2Time8 = MinHTG2Time8
    ElseIf NormHTG2Time8 > MaxHTG2Time8 Then
        NormHTG2Time8 = MaxHTG2Time8
```

```
            End If
            If NormHTG2Time8 < 1 Then
                EndHTG2Date = EndwarehouseDate + 1
            Else
                EndHTG2Date = EndwarehouseDate + Int(NormHTG2Time8)
            End If
        ElseIf OrderClass = 12 Then
            NormHTG2Time12 = WorksheetFunction.Norm_Inv(Rnd(), AvgHTG2Time12, DevHTG2Time12)
            If NormHTG2Time12 < MinHTG2Time12 Then
                NormHTG2Time12 = MinHTG2Time12
            ElseIf NormHTG2Time12 > MaxHTG2Time12 Then
                NormHTG2Time12 = MaxHTG2Time12
            End If
            If NormHTG2Time12 < 1 Then
                EndHTG2Date = EndwarehouseDate + 1
            Else
                EndHTG2Date = EndwarehouseDate + Int(NormHTG2Time12)
            End If
        End If
        If WorksheetFunction.Weekday(CDate(EndHTG2Date)) = 1 Then
            EndHTG2Date = EndHTG2Date + 1
        ElseIf WorksheetFunction.Weekday(CDate(EndHTG2Date)) = 7 Then
            EndHTG2Date = EndHTG2Date + 2
        End If
        If EndHTG2Date < today Then
        sh2.Cells(j + 1, 9) = CDate(EndHTG2Date)
        End If

        If OrderClass = 8 Then                                                          'Production HTG3 time
            NormHTG3Time8 = WorksheetFunction.Norm_Inv(Rnd(), AvgHTG3Time8, DevHTG3Time8)
            If NormHTG3Time8 < MinHTG3Time8 Then
                NormHTG3Time8 = MinHTG3Time8
            ElseIf NormHTG3Time8 > MaxHTG3Time8 Then
                NormHTG3Time8 = MaxHTG3Time8
            End If
            If NormHTG3Time8 < 1 Then
                EndHTG3Date = EndHTG2Date + 1
            Else
                EndHTG3Date = EndHTG2Date + Int(NormHTG3Time8)
            End If
            If WorksheetFunction.Weekday(CDate(EndHTG3Date)) = 1 Then
                EndHTG3Date = EndHTG3Date + 1
            ElseIf WorksheetFunction.Weekday(CDate(EndHTG3Date)) = 7 Then
                EndHTG3Date = EndHTG3Date + 2
            End If

            If EndHTG2Date < today Then
                sh2.Cells(j + 1, 10) = CDate(EndHTG3Date)
            End If

        ElseIf OrderClass = 12 Then
            NormHTG3Time12 = WorksheetFunction.Norm_Inv(Rnd(), AvgHTG3Time12, DevHTG3Time12)

            If NormHTG3Time12 < MinHTG3Time12 Then
                NormHTG3Time12 = MinHTG3Time12
            ElseIf NormHTG3Time12 > MaxHTG3Time12 Then
                NormHTG3Time12 = MaxHTG3Time12
            End If

            If NormHTG3Time12 < 1 Then
                EndHTG3Date = EndHTG2Date + 1
            Else
                EndHTG3Date = EndHTG2Date + Int(NormHTG3Time12)
            End If

            If WorksheetFunction.Weekday(CDate(EndHTG3Date)) = 1 Then
                EndHTG3Date = EndHTG3Date + 1
            ElseIf WorksheetFunction.Weekday(CDate(EndHTG3Date)) = 7 Then
                EndHTG3Date = EndHTG3Date + 2
            End If

            If EndHTG2Date < today Then
                sh2.Cells(j + 1, 10) = CDate(EndHTG3Date)
            End If
        End If

        NormOrderFinishTime = WorksheetFunction.Norm_Inv(Rnd(), AvgOrderFinishTime, DevOrderFinishTime)     'Order finish time
        If NormOrderFinishTime < MinOrderFinishTime Then
            NormOrderFinishTime = MinOrderFinishTime
        ElseIf NormOrderFinishTime > MaxOrderFinishTime Then
            NormOrderFinishTime = MaxOrderFinishTime
        End If

        If OrderClass = 4 Then
            If NormOrderFinishTime < 1 Then
                OrderFinishDate = EndHTG2Date + 1
            Else
                OrderFinishDate = EndHTG2Date + Int(NormOrderFinishTime)
            End If
        Else
            If NormOrderFinishTime < 1 Then
                OrderFinishDate = EndHTG3Date + 1
            Else
                OrderFinishDate = EndHTG3Date + Int(NormOrderFinishTime)
            End If
        End If
        If WorksheetFunction.Weekday(CDate(OrderFinishDate)) = 1 Then
            OrderFinishDate = OrderFinishDate + 1
        ElseIf WorksheetFunction.Weekday(CDate(OrderFinishDate)) = 7 Then
            OrderFinishDate = OrderFinishDate + 2
        End If

        If OrderFinishDate < today Then
            sh2.Cells(j + 1, 11) = CDate(OrderFinishDate)
        End If

        j = j + 1

    Next k
    End If

Next i

End Sub
```

Appendix G.2: macro generating capacity data

```
Sub Capacityhours()

Dim sh1 As Worksheet
Dim sh2 As Worksheet
Dim sh3 As Worksheet

Set sh1 = ActiveWorkbook.Sheets("Input dummy data")
Set sh2 = ActiveWorkbook.Sheets("Eventlog")
Set sh3 = ActiveWorkbook.Sheets("Capacity")

sh3.Cells.Clear
sh3.Cells(1, 1) = "Weeknr"
sh3.Cells(1, 2) = "Start"
sh3.Cells(1, 3) = "End"
sh3.Cells(1, 4) = "Projects"
sh3.Cells(1, 5) = "Work prep."
sh3.Cells(1, 6) = "Warehouse"
sh3.Cells(1, 7) = "HTG2"
sh3.Cells(1, 8) = "HTG3"
sh3.Cells(1, 9) = "Shipping prep."

Dim startdate As Long
Dim i, j As Long
Dim today As Long
Dim week As Long
Dim normProjects, avgProjects, devProjects, minProjects, maxProjects As Double
Dim normWorkPrep, avgWorkPrep, devWorkPrep, minWorkPrep, maxWorkPrep As Double
Dim normWarehouse, avgWarehouse, devWarehouse, minWarehouse, maxWarehouse As Double
Dim normHTG2, avgHTG2, devHTG2, minHTG2, maxHTG2 As Double
Dim normHTG3, avgHTG3, devHTG3, minHTG3, maxHTG3 As Double
Dim normShipping, avgShipping, devShipping, minShipping, maxShipping As Double

avgProjects = sh1.Cells(21, 3)
devProjects = sh1.Cells(21, 4)
minProjects = sh1.Cells(21, 5)
maxProjects = sh1.Cells(21, 6)
avgWorkPrep = sh1.Cells(22, 3)
devWorkPrep = sh1.Cells(22, 4)
minWorkPrep = sh1.Cells(22, 5)
maxWorkPrep = sh1.Cells(22, 6)
avgWarehouse = sh1.Cells(23, 3)
devWarehouse = sh1.Cells(23, 4)
minWarehouse = sh1.Cells(23, 5)
maxWarehouse = sh1.Cells(23, 6)
avgHTG2 = sh1.Cells(24, 3)
devHTG2 = sh1.Cells(24, 4)
minHTG2 = sh1.Cells(24, 5)
maxHTG2 = sh1.Cells(24, 6)
avgHTG3 = sh1.Cells(25, 3)
devHTG3 = sh1.Cells(25, 4)
minHTG3 = sh1.Cells(25, 5)
maxHTG3 = sh1.Cells(25, 6)
avgShipping = sh1.Cells(26, 3)
devShipping = sh1.Cells(26, 4)
minShipping = sh1.Cells(26, 5)
maxShipping = sh1.Cells(26, 6)

startdate = sh1.Cells(2, 4)
today = sh1.Cells(3, 4)
j = 1
For i = startdate To today + 70

If i = startdate Then
    If WorksheetFunction.Weekday(CDate(i)) <> 2 Then
        If WorksheetFunction.Weekday(CDate(i)) = 1 Then
            sh3.Cells(j + 1, 2) = CDate(i - 6)
            sh3.Cells(j + 1, 1) = WorksheetFunction.WeekNum(CDate(i - 6)) - 1
            sh3.Cells(j + 1, 3) = CDate(i)
        Else
            sh3.Cells(j + 1, 2) = CDate(i - WorksheetFunction.Weekday(CDate(i)) + 2)
            sh3.Cells(j + 1, 1) = WorksheetFunction.WeekNum(sh3.Cells(j + 1, 2)) - 1
            sh3.Cells(j + 1, 3) = CDate(i + 8 - WorksheetFunction.Weekday(CDate(i)))
        End If
        normProjects = WorksheetFunction.Norm_Inv(Rnd(), avgProjects, devProjects)
        If normProjects < minProjects Then
            normProjects = minProjects
        ElseIf normProjects > maxProjects Then
            normProjects = maxProjects
        End If
        normWorkPrep = WorksheetFunction.Norm_Inv(Rnd(), avgWorkPrep, devWorkPrep)
        If normWorkPrep < minWorkPrep Then
            normWorkPrep = minWorkPrep
        ElseIf normWorkPrep > maxWorkPrep Then
            normWorkPrep = maxWorkPrep
        End If
        normWarehouse = WorksheetFunction.Norm_Inv(Rnd(), avgWarehouse, devWarehouse)
        If normWarehouse < minWarehouse Then
            normWarehouse = minWarehouse
        ElseIf normWarehouse > maxWarehouse Then
            normWarehouse = maxWarehouse
        End If
        normHTG2 = WorksheetFunction.Norm_Inv(Rnd(), avgHTG2, devHTG2)
        If normHTG2 < minHTG2 Then
            normHTG2 = minHTG2
        ElseIf normHTG2 > maxHTG2 Then
            normHTG2 = maxHTG2
        End If
        normHTG3 = WorksheetFunction.Norm_Inv(Rnd(), avgHTG3, devHTG3)
        If normHTG3 < minHTG3 Then
            normHTG3 = minHTG3
        ElseIf normHTG3 > maxHTG3 Then
            normHTG3 = maxHTG3
        End If
        normShipping = WorksheetFunction.Norm_Inv(Rnd(), avgShipping, devShipping)
        If normShipping < minShipping Then
            normShipping = minShipping
        ElseIf normShipping > maxShipping Then
```

```
            normShipping = maxShipping
        End If
        sh3.Cells(j + 1, 4) = Int(normProjects)
        sh3.Cells(j + 1, 5) = Int(normWorkPrep)
        sh3.Cells(j + 1, 6) = Int(normWarehouse)
        sh3.Cells(j + 1, 7) = Int(normHTG2)
        sh3.Cells(j + 1, 8) = Int(normHTG3)
        sh3.Cells(j + 1, 9) = Int(normShipping)
        j = j + 1

    End If
End If
If WorksheetFunction.Weekday(CDate(i)) = 2 Then
    sh3.Cells(j + 1, 2) = CDate(i)
    sh3.Cells(j + 1, 1) = WorksheetFunction.WeekNum(CDate(i)) - 1
    sh3.Cells(j + 1, 3) = CDate(i + 6)
    normProjects = WorksheetFunction.Norm_Inv(Rnd(), avgProjects, devProjects)
    If normProjects < minProjects Then
        normProjects = minProjects
    ElseIf normProjects > maxProjects Then
        normProjects = maxProjects
    End If
    normWorkPrep = WorksheetFunction.Norm_Inv(Rnd(), avgWorkPrep, devWorkPrep)
    If normWorkPrep < minWorkPrep Then
        normWorkPrep = minWorkPrep
    ElseIf normWorkPrep > maxWorkPrep Then
        normWorkPrep = maxWorkPrep
    End If
    normWarehouse = WorksheetFunction.Norm_Inv(Rnd(), avgWarehouse, devWarehouse)
    If normWarehouse < minWarehouse Then
        normWarehouse = minWarehouse
    ElseIf normWarehouse > maxWarehouse Then
        normWarehouse = maxWarehouse
    End If
    normHTG2 = WorksheetFunction.Norm_Inv(Rnd(), avgHTG2, devHTG2)
    If normHTG2 < minHTG2 Then
        normHTG2 = minHTG2
    ElseIf normHTG2 > maxHTG2 Then
        normHTG2 = maxHTG2
    End If
    normHTG3 = WorksheetFunction.Norm_Inv(Rnd(), avgHTG3, devHTG3)
    If normHTG3 < minHTG3 Then
        normHTG3 = minHTG3
    ElseIf normHTG3 > maxHTG3 Then
        normHTG3 = maxHTG3
    End If
    normShipping = WorksheetFunction.Norm_Inv(Rnd(), avgShipping, devShipping)
    If normShipping < minShipping Then
        normShipping = minShipping
    ElseIf normShipping > maxShipping Then
        normShipping = maxShipping
    End If
    sh3.Cells(j + 1, 4) = Int(normProjects)
    sh3.Cells(j + 1, 5) = Int(normWorkPrep)
    sh3.Cells(j + 1, 6) = Int(normWarehouse)
    sh3.Cells(j + 1, 7) = Int(normHTG2)
    sh3.Cells(j + 1, 8) = Int(normHTG3)
    sh3.Cells(j + 1, 9) = Int(normShipping)
    j = j + 1
End If
Next i

End Sub
```

Appendix G.3: macro simulate one day

```
Sub SimulateOneDay()
Dim sh1 As Worksheet
Dim sh2 As Worksheet
Dim sh3 As Worksheet
Set sh1 = ActiveWorkbook.Sheets("Input dummy data")
Set sh2 = ActiveWorkbook.Sheets("Eventlog")
Set sh3 = ActiveWorkbook.Sheets("Capacity")


Dim i, j, k, a As Long
Dim startdate As Long
Dim today As Long
Dim Ordernr As Long
Dim AvgNumOrdDay, DevNumOrdDay, MinNumOrdDay, MaxNumOrdDay As Double
Dim AvgProTime, DevProTime, MinProTime, MaxProTime, NormProTime As Double
Dim AvgWorkPrepTime, DevWorkPrepTime, MinWorkPrepTime, MaxWorkPrepTime, NormWorkPrepTime As Double
Dim AvgWarehouseTime, DevWarehouseTime, MinWarehouseTime, MaxWarehouseTime, NormWarehouseTime As Double
Dim AvgHTG2Time4, DevHTG2Time4, MinHTG2Time4, MaxHTG2Time4, NormHTG2Time4 As Double
Dim AvgHTG2Time8, DevHTG2Time8, MinHTG2Time8, MaxHTG2Time8, NormHTG2Time8 As Double
Dim AvgHTG2Time12, DevHTG2Time12, MinHTG2Time12, MaxHTG2Time12, NormHTG2Time12 As Double
Dim AvgHTG3Time8, DevHTG3Time8, MinHTG3Time8, MaxHTG3Time8, NormHTG3Time8 As Double
Dim AvgHTG3Time12, DevHTG3Time12, MinHTG3Time12, MaxHTG3Time12, NormHTG3Time12 As Double
Dim AvgOrderFinishTime, DevOrderFinishTime, MinOrderFinishTime, MaxOrderFinishTime, NormOrderFinishTime As Double


Dim dblNumberOrders As Double
Dim intNumberOrders As Long
Dim OrderClass As Integer
Dim rndOrderClass As Double
Dim rndWishedDate As Integer
Dim Weekday As Integer
Dim WeekdayWishedDate As Long
Dim ProjectsTime As Long
Dim EndProjectsDate As Long
Dim WorkprepTime As Long
Dim EndWorkprepDate As Long
Dim EndwarehouseDate As Long
Dim EndHTG2Date As Long
Dim EndHTG3Date As Long
Dim OrderFinishDate As Long
Dim ColumnNr, totalrows As Long

Dim OrderReleaseDate1, OrderReleaseDate2, OrderReleaseDate3 As Long
Dim WeekdayORD1, WeekdayORD2, WeekdayORD3 As Long
Dim rand1, rand2, rand3, rand4, rand5 As Long


'get variables from input sheet
startdate = sh1.Cells(2, 4)
today = sh1.Cells(3, 4) + 1
sh1.Cells(3, 3) = CDate(today)
AvgNumOrdDay = sh1.Cells(9, 3)
DevNumOrdDay = sh1.Cells(9, 4)
MinNumOrdDay = sh1.Cells(9, 5)
MaxNumOrdDay = sh1.Cells(9, 6)
AvgProTime = sh1.Cells(10, 3)
DevProTime = sh1.Cells(10, 4)
MinProTime = sh1.Cells(10, 5)
MaxProTime = sh1.Cells(10, 6)
AvgWorkPrepTime = sh1.Cells(11, 3)
DevWorkPrepTime = sh1.Cells(11, 4)
MinWorkPrepTime = sh1.Cells(11, 5)
MaxWorkPrepTime = sh1.Cells(11, 6)
AvgWarehouseTime = sh1.Cells(12, 3)
DevWarehouseTime = sh1.Cells(12, 4)
MinWarehouseTime = sh1.Cells(12, 5)
MaxWarehouseTime = sh1.Cells(12, 6)
AvgHTG2Time4 = sh1.Cells(13, 3)
DevHTG2Time4 = sh1.Cells(13, 4)
MinHTG2Time4 = sh1.Cells(13, 5)
MaxHTG2Time4 = sh1.Cells(13, 6)
AvgHTG2Time8 = sh1.Cells(14, 3)
DevHTG2Time8 = sh1.Cells(14, 4)
MinHTG2Time8 = sh1.Cells(14, 5)
MaxHTG2Time8 = sh1.Cells(14, 6)
AvgHTG2Time12 = sh1.Cells(15, 3)
DevHTG2Time12 = sh1.Cells(15, 4)
MinHTG2Time12 = sh1.Cells(15, 5)
MaxHTG2Time12 = sh1.Cells(15, 6)
AvgHTG3Time8 = sh1.Cells(16, 3)
DevHTG3Time8 = sh1.Cells(16, 4)
MinHTG3Time8 = sh1.Cells(16, 5)
MaxHTG3Time8 = sh1.Cells(16, 6)
AvgHTG3Time12 = sh1.Cells(17, 3)
DevHTG3Time12 = sh1.Cells(17, 4)
MinHTG3Time12 = sh1.Cells(17, 5)
MaxHTG3Time12 = sh1.Cells(17, 6)
AvgOrderFinishTime = sh1.Cells(18, 3)
DevOrderFinishTime = sh1.Cells(18, 4)
MinOrderFinishTime = sh1.Cells(18, 5)
MaxOrderFinishTime = sh1.Cells(18, 6)
Dim week As Long
Dim normProjects, avgProjects, devProjects, minProjects, maxProjects As Double
Dim normWorkPrep, avgWorkPrep, devWorkPrep, minWorkPrep, maxWorkPrep As Double
Dim normWarehouse, avgWarehouse, devWarehouse, minWarehouse, maxWarehouse As Double
Dim normHTG2, avgHTG2, devHTG2, minHTG2, maxHTG2 As Double
Dim normHTG3, avgHTG3, devHTG3, minHTG3, maxHTG3 As Double
Dim normShipping, avgShipping, devShipping, minShipping, maxShipping As Double

avgProjects = sh1.Cells(21, 3)
devProjects = sh1.Cells(21, 4)
minProjects = sh1.Cells(21, 5)
maxProjects = sh1.Cells(21, 6)
avgWorkPrep = sh1.Cells(22, 3)
devWorkPrep = sh1.Cells(22, 4)
minWorkPrep = sh1.Cells(22, 5)
maxWorkPrep = sh1.Cells(22, 6)
```

```
avgWarehouse = sh1.Cells(23, 3)
devWarehouse = sh1.Cells(23, 4)
minWarehouse = sh1.Cells(23, 5)
maxWarehouse = sh1.Cells(23, 6)
avgHTG2 = sh1.Cells(24, 3)
devHTG2 = sh1.Cells(24, 4)
minHTG2 = sh1.Cells(24, 5)
maxHTG2 = sh1.Cells(24, 6)
avgHTG3 = sh1.Cells(25, 3)
devHTG3 = sh1.Cells(25, 4)
minHTG3 = sh1.Cells(25, 5)
maxHTG3 = sh1.Cells(25, 6)
avgShipping = sh1.Cells(26, 3)
devShipping = sh1.Cells(26, 4)
minShipping = sh1.Cells(26, 5)
maxShipping = sh1.Cells(26, 6)

i = today + 70
a = sh3.Cells(sh3.Rows.count, 1).End(xlUp).Row 'Update capacity if today is monday
If WorksheetFunction.Weekday(CDate(i)) = 2 Then
    sh3.Cells(a + 1, 2) = CDate(i)
    sh3.Cells(a + 1, 1) = WorksheetFunction.WeekNum(CDate(i)) - 1
    sh3.Cells(a + 1, 3) = CDate(i + 6)
    normProjects = WorksheetFunction.Norm_Inv(Rnd(), avgProjects, devProjects)
    If normProjects < minProjects Then
        normProjects = minProjects
    ElseIf normProjects > maxProjects Then
         normProjects = maxProjects
    End If
    normWorkPrep = WorksheetFunction.Norm_Inv(Rnd(), avgWorkPrep, devWorkPrep)
    If normWorkPrep < minWorkPrep Then
        normWorkPrep = minWorkPrep
    ElseIf normWorkPrep > maxWorkPrep Then
        normWorkPrep = maxWorkPrep
    End If
    normWarehouse = WorksheetFunction.Norm_Inv(Rnd(), avgWarehouse, devWarehouse)
     If normWarehouse < minWarehouse Then
        normWarehouse = minWarehouse
    ElseIf normWarehouse > maxWarehouse Then
        normWarehouse = maxWarehouse
    End If
    normHTG2 = WorksheetFunction.Norm_Inv(Rnd(), avgHTG2, devHTG2)
    If normHTG2 < minHTG2 Then
        normHTG2 = minHTG2
    ElseIf normHTG2 > maxHTG2 Then
        normHTG2 = maxHTG2
    End If
    normHTG3 = WorksheetFunction.Norm_Inv(Rnd(), avgHTG3, devHTG3)
    If normHTG3 < minHTG3 Then
        normHTG3 = minHTG3
     ElseIf normHTG3 > maxHTG3 Then
        normHTG3 = maxHTG3
     End If
    normShipping = WorksheetFunction.Norm_Inv(Rnd(), avgShipping, devShipping)
    If normShipping < minShipping Then
        normShipping = minShipping
    ElseIf normShipping > maxShipping Then
        normShipping = maxShipping
    End If
    sh3.Cells(a + 1, 4) = Int(normProjects)
    sh3.Cells(a + 1, 5) = Int(normWorkPrep)
    sh3.Cells(a + 1, 6) = Int(normWarehouse)
    sh3.Cells(a + 1, 7) = Int(normHTG2)
    sh3.Cells(a + 1, 8) = Int(normHTG3)
    sh3.Cells(a + 1, 9) = Int(normShipping)
    a = a + 1
End If


totalrows = sh2.Cells(sh2.Rows.count, 1).End(xlUp).Row + 1              'update eventlog
For i = 2 To totalrows
ColumnNr = sh2.Cells(i, sh2.Columns.count).End(xlToLeft).Column
If ColumnNr = 1 Then
Weekday = WorksheetFunction.Weekday(CDate(today))
    If Weekday < 7 And Weekday > 1 Then 'orders worden niet in het weekend ingevoerd
        dblNumberOrders = WorksheetFunction.Norm_Inv(Rnd(), AvgNumOrdDay, DevNumOrdDay)
        If dblNumberOrders < MinNumOrdDay Then
            dblNumberOrders = MinNumOrdDay
        ElseIf dblNumberOrders > MaxNumOrdDay Then
            dblNumberOrders = MaxNumOrdDay
        End If

        intNumberOrders = Int(dblNumberOrders) 'bepalen aantal orders per dag gebasseerd op interviews
        If intNumberOrders < 1 Then 'het aantal orders is minimaal 1
        intNumberOrders = 1
        End If
        j = totalrows
        For k = 1 To intNumberOrders
            sh2.Cells(j, 1) = "400" + j 'ordernummer maken

            rndOrderClass = Rnd() 'determine order class with rates 60/30/10
        If rndOrderClass < 0.6 Then
            OrderClass = 4
        ElseIf rndOrderClass < 0.9 Then
            OrderClass = 8
        Else
            OrderClass = 12
        End If
        sh2.Cells(j, 2) = OrderClass
        sh2.Cells(j, 4) = CDate(today) 'invullen datum invoer
        rand1 = Rnd()
        If Int(WorksheetFunction.Norm_Inv(rand1, 5, 3)) < 0 Then
        'klanten willen in de meeste gevallen hun order op een specifieke dag
        'er is uitgegaan van 5 weken als gemiddelde.
            rndWishedDate = 1
        Else
            rndWishedDate = Int(WorksheetFunction.Norm_Inv(rand1, 5, 3)) + 1
```

```
        End If
        rand2 = Rnd()
        If WorksheetFunction.Weekday(CDate(today + (rndWishedDate + OrderClass - 1) * 7 + Int(rand2 * 7 + 1))) = 7 Then
        'ook weer in het weekend levert prinzen niet
            sh2.Cells(j, 3) = CDate(today + (rndWishedDate + OrderClass - 1) * 7 + Int(rand2 * 7))
        ElseIf WorksheetFunction.Weekday(CDate(today + (rndWishedDate + OrderClass - 1) * 7 + Int(rand2 * 7 + 1))) = 1 Then
            sh2.Cells(j, 3) = CDate(today + (rndWishedDate + OrderClass - 1) * 7 + Int(rand2 * 7 - 1))
        Else
            sh2.Cells(j, 3) = CDate(today + (rndWishedDate + OrderClass - 1) * 7 + Int(rand2 * 7 + 1))
        End If
        j = j + 1
        Next k
        End If
ElseIf ColumnNr = 4 Then
    NormProTime = WorksheetFunction.Norm_Inv(Rnd(), AvgProTime, DevProTime) 'projects time
        If NormProTime < MinProTime Then
            NormProTime = MinProTime
        ElseIf NormProTime > MaxProTime Then
            NormProTime = MaxProTime
        End If
        ProjectsTime = Int(NormProTime * sh2.Cells(i, 2) / 4)
        If ProjectsTime < 1 Then
            EndProjectsDate = CLng(Cells(i, ColumnNr)) + 1
        Else
            EndProjectsDate = CLng(Cells(i, ColumnNr)) + ProjectsTime
        End If
        If WorksheetFunction.Weekday(CDate(EndProjectsDate)) = 1 Then
            EndProjectsDate = EndProjectsDate + 1
        ElseIf WorksheetFunction.Weekday(CDate(EndProjectsDate)) = 7 Then
            EndProjectsDate = EndProjectsDate + 2
        End If

        If EndProjectsDate < today Then  'let op dit stukje
            sh2.Cells(i, 5) = CDate(today)
        End If



ElseIf ColumnNr = 5 Then 'overal toevoegen stukje
    NormWorkPrepTime = WorksheetFunction.Norm_Inv(Rnd(), AvgWorkPrepTime, DevWorkPrepTime) 'work preperation time
        If NormWorkPrepTime < MinWorkPrepTime Then
        NormWorkPrepTime = MinWorkPrepTime
        ElseIf NormWorkPrepTime > MaxWorkPrepTime Then
        NormWorkPrepTime = MaxWorkPrepTime
        End If
        WorkprepTime = Int(NormWorkPrepTime * sh2.Cells(i, 2) / 4)
        If WorkprepTime < 1 Then
            EndWorkprepDate = CLng(sh2.Cells(i, 5)) + 1
        Else
            EndWorkprepDate = CLng(sh2.Cells(i, 5)) + WorkprepTime
        End If

        If WorksheetFunction.Weekday(CDate(EndWorkprepDate)) = 1 Then
            EndWorkprepDate = EndWorkprepDate + 1
        ElseIf WorksheetFunction.Weekday(CDate(EndWorkprepDate)) = 7 Then
            EndWorkprepDate = EndWorkprepDate + 2
        End If
        If EndWorkprepDate < today Then
            sh2.Cells(i, 6) = CDate(today)
        OrderReleaseDate1 = CLng(sh2.Cells(i, 3)) - 3                    'three days for packing
        WeekdayORD1 = WorksheetFunction.Weekday(CDate(OrderReleaseDate1))
        If WeekdayORD1 = 1 Or WeekdayORD1 = 7 Or WeekdayORD1 = 6 Then
            OrderReleaseDate1 = OrderReleaseDate1 - 2
        End If

        If sh2.Cells(i, 2) = 4 Then                                              'days for assembly
            OrderReleaseDate2 = OrderReleaseDate1 - 14
        ElseIf sh2.Cells(i, 2) = 8 Then
            OrderReleaseDate2 = OrderReleaseDate1 - 28
        ElseIf sh2.Cells(i, 2) = 12 Then
            OrderReleaseDate2 = OrderReleaseDate1 - 56
        End If

        OrderReleaseDate3 = OrderReleaseDate2 - 2                          'two days for order picking
        WeekdayORD3 = WorksheetFunction.Weekday(CDate(OrderReleaseDate2))
        If WeekdayORD3 = 1 Or WeekdayORD3 = 7 Then
            OrderReleaseDate3 = OrderReleaseDate3 - 2
        End If
        sh2.Cells(i, 12) = CDate(OrderReleaseDate3)
        End If
ElseIf ColumnNr = 12 Then
    If sh2.Cells(i, 7) = "" Then
        If CLng(sh2.Cells(i, 12)) = today Then
            sh2.Cells(i, 7) = CDate(today)
        End If
    ElseIf sh2.Cells(i, 8) = "" Then
    NormWarehouseTime = WorksheetFunction.Norm_Inv(Rnd(), AvgWarehouseTime, DevWarehouseTime) 'warehouse time
        If NormWarehouseTime < MinWarehouseTime Then
            NormWarehouseTime = MinWarehouseTime
        ElseIf NormWarehouseTime > MaxWarehouseTime Then
            NormWarehouseTime = MaxWarehouseTime
        End If

        If NormWarehouseTime < 1 Then
            EndwarehouseDate = CLng(Cells(i, 7)) + 1
        Else
            EndwarehouseDate = CLng(Cells(i, 7)) + Int(NormWarehouseTime)
        End If

        If WorksheetFunction.Weekday(CDate(EndwarehouseDate)) = 1 Then
            EndwarehouseDate = EndwarehouseDate + 1
        ElseIf WorksheetFunction.Weekday(CDate(EndwarehouseDate)) = 7 Then
            EndwarehouseDate = EndwarehouseDate + 2
        End If
```

```
    If EndwarehouseDate < today Then
        sh2.Cells(i, 8) = CDate(today)
    End If
ElseIf sh2.Cells(i, 9) = "" Then
If sh2.Cells(i, 2) = 4 Then                                                    'htg2 time
        NormHTG2Time4 = WorksheetFunction.Norm_Inv(Rnd(), AvgHTG2Time4, DevHTG2Time4)
        If NormHTG2Time4 < MinHTG2Time4 Then
            NormHTG2Time4 = MinHTG2Time4
        ElseIf NormHTG2Time4 > MaxHTG2Time4 Then
            NormHTG2Time4 = MaxHTG2Time4
        End If
        If NormHTG2Time4 < 1 Then
            EndHTG2Date = CLng(sh2.Cells(i, ColumnNr)) + 1
        Else
            EndHTG2Date = CLng(sh2.Cells(i, ColumnNr)) + Int(NormHTG2Time4)
        End If
    ElseIf sh2.Cells(i, 2) = 8 Then
        NormHTG2Time8 = WorksheetFunction.Norm_Inv(Rnd(), AvgHTG2Time8, DevHTG2Time8)
        If NormHTG2Time8 < MinHTG2Time8 Then
            NormHTG2Time8 = MinHTG2Time8
        ElseIf NormHTG2Time8 > MaxHTG2Time8 Then
            NormHTG2Time8 = MaxHTG2Time8
        End If
        If NormHTG2Time8 < 1 Then
            EndHTG2Date = CLng(sh2.Cells(i, ColumnNr)) + 1
        Else
            EndHTG2Date = CLng(sh2.Cells(i, ColumnNr)) + Int(NormHTG2Time8)
        End If
    ElseIf sh2.Cells(i, 2) = 12 Then
        NormHTG2Time12 = WorksheetFunction.Norm_Inv(Rnd(), AvgHTG2Time12, DevHTG2Time12)
        If NormHTG2Time12 < MinHTG2Time12 Then
            NormHTG2Time12 = MinHTG2Time12
        ElseIf NormHTG2Time12 > MaxHTG2Time12 Then
            NormHTG2Time12 = MaxHTG2Time12
        End If
        If NormHTG2Time12 < 1 Then
            EndHTG2Date = CLng(sh2.Cells(i, ColumnNr)) + 1
        Else
            EndHTG2Date = CLng(sh2.Cells(i, ColumnNr)) + Int(NormHTG2Time12)
        End If
    End If
    If WorksheetFunction.Weekday(CDate(EndHTG2Date)) = 1 Then
        EndHTG2Date = EndHTG2Date + 1
    ElseIf WorksheetFunction.Weekday(CDate(EndHTG2Date)) = 7 Then
        EndHTG2Date = EndHTG2Date + 2
    End If
    If EndHTG2Date < today Then
    sh2.Cells(i, 9) = CDate(today)
    End If
ElseIf sh2.Cells(i, 10) = "" Then
 If sh2.Cells(i, 2) = 8 Then                                                   'HTG3 time
        NormHTG3Time8 = WorksheetFunction.Norm_Inv(Rnd(), AvgHTG3Time8, DevHTG3Time8)
        If NormHTG3Time8 < MinHTG3Time8 Then
            NormHTG3Time8 = MinHTG3Time8
        ElseIf NormHTG3Time8 > MaxHTG3Time8 Then
            NormHTG3Time8 = MaxHTG3Time8
        End If
        If NormHTG3Time8 < 1 Then
            EndHTG3Date = CLng(sh2.Cells(i, ColumnNr)) + 1
        Else
            EndHTG3Date = CLng(sh2.Cells(i, ColumnNr)) + Int(NormHTG3Time8)
        End If
        If WorksheetFunction.Weekday(CDate(EndHTG3Date)) = 1 Then
            EndHTG3Date = EndHTG3Date + 1
        ElseIf WorksheetFunction.Weekday(CDate(EndHTG3Date)) = 7 Then
            EndHTG3Date = EndHTG3Date + 2
        End If

        If EndHTG3Date < today Then
            sh2.Cells(i, 10) = CDate(today)
        End If

    ElseIf sh2.Cells(i, 2) = 12 Then
        NormHTG3Time12 = WorksheetFunction.Norm_Inv(Rnd(), AvgHTG3Time12, DevHTG3Time12)

        If NormHTG3Time12 < MinHTG3Time12 Then
            NormHTG3Time12 = MinHTG3Time12
        ElseIf NormHTG3Time12 > MaxHTG3Time12 Then
            NormHTG3Time12 = MaxHTG3Time12
        End If

        If NormHTG3Time12 < 1 Then
            EndHTG3Date = CLng(sh2.Cells(i, ColumnNr)) + 1
        Else
            EndHTG3Date = CLng(sh2.Cells(i, ColumnNr)) + Int(NormHTG3Time12)
        End If

        If WorksheetFunction.Weekday(CDate(EndHTG3Date)) = 1 Then
            EndHTG3Date = EndHTG3Date + 1
        ElseIf WorksheetFunction.Weekday(CDate(EndHTG3Date)) = 7 Then
            EndHTG3Date = EndHTG3Date + 2
        End If
        If EndHTG3Date < today Then
            sh2.Cells(i, 10) = CDate(today)
        End If

     ElseIf sh2.Cells(i, 2) = 4 Then
        If NormOrderFinishTime < 1 Then
            OrderFinishDate = CLng(sh2.Cells(i, 9)) + 1
        Else
            OrderFinishDate = CLng(sh2.Cells(i, 9)) + Int(NormOrderFinishTime)
        End If
    If WorksheetFunction.Weekday(CDate(OrderFinishDate)) = 1 Then
        OrderFinishDate = OrderFinishDate + 1
    ElseIf WorksheetFunction.Weekday(CDate(OrderFinishDate)) = 7 Then
        OrderFinishDate = OrderFinishDate + 2
    End If
    If OrderFinishDate < today Then
```

```
            sh2.Cells(i, 11) = CDate(today)
        End If

        End If
    ElseIf sh2.Cells(i, 11) = "" Then
     NormOrderFinishTime = WorksheetFunction.Norm_Inv(Rnd(), AvgOrderFinishTime, DevOrderFinishTime)     'Order finish time
        If NormOrderFinishTime < MinOrderFinishTime Then
            NormOrderFinishTime = MinOrderFinishTime
        ElseIf NormOrderFinishTime > MaxOrderFinishTime Then
            NormOrderFinishTime = MaxOrderFinishTime
        End If

        If NormOrderFinishTime < 1 Then
            OrderFinishDate = CLng(sh2.Cells(i, ColumnNr)) + 1
        Else
            OrderFinishDate = CLng(sh2.Cells(i, ColumnNr)) + Int(NormOrderFinishTime)
        End If
        If WorksheetFunction.Weekday(CDate(OrderFinishDate)) = 1 Then
            OrderFinishDate = OrderFinishDate + 1
        ElseIf WorksheetFunction.Weekday(CDate(OrderFinishDate)) = 7 Then
            OrderFinishDate = OrderFinishDate + 2
        End If
        If OrderFinishDate < today Then
            sh2.Cells(i, 11) = CDate(today)
        End If
    End If
End If

Next i


End Sub
```

Appendix G.4: KPI average time per phase macro

```
Sub KPIaverageTimePerPhase()
Dim sh1 As Worksheet
Dim sh2 As Worksheet
Dim sh3 As Worksheet
Dim sh4 As Worksheet


Set sh1 = ActiveWorkbook.Sheets("Input dummy data")
Set sh2 = ActiveWorkbook.Sheets("Eventlog")
Set sh3 = ActiveWorkbook.Sheets("Capacity")
Set sh4 = ActiveWorkbook.Sheets("KPI average time per phase")

Dim i, j, a, b As Long
Dim NumberRows, NumberRows1 As Long
Dim today As Long
Dim sum, count As Long

NumberRows = sh2.Cells(sh2.Rows.count, 11).End(xlUp).Row
today = CLng(sh1.Cells(3, 3))
sh4.Cells.Clear
sh4.Cells(1, 1) = "Projects"
sh4.Cells(1, 2) = "Work preperation"
sh4.Cells(1, 3) = "Warehouse"
sh4.Cells(1, 4) = "Production HTG2"
sh4.Cells(1, 5) = "Production HTG3"
sh4.Cells(1, 6) = "Shipping preperation"
sh4.Cells(1, 8) = "Projects average"
sh4.Cells(1, 9) = "Work preperation average"
sh4.Cells(1, 10) = "warehouse average"
sh4.Cells(1, 11) = "Production HTG2 average"
sh4.Cells(1, 12) = "Production HTG3 average"
sh4.Cells(1, 13) = "Shipping preperation average"
j = 2
For i = 2 To NumberRows

    If CLng(sh2.Cells(i, 11)) <= today Or CLng(sh2.Cells(i, 11)) >= today - 7 Then
        If CLng(sh2.Cells(i, 11)) > 0 Then
            sh4.Cells(j, 1) = Int((CLng(sh2.Cells(i, 5)) - CLng(sh2.Cells(i, 4))) * 4 / sh2.Cells(i, 2) - 0.001) + 1
            sh4.Cells(j, 2) = Int((CLng(sh2.Cells(i, 6)) - CLng(sh2.Cells(i, 5))) * 4 / sh2.Cells(i, 2) - 0.001) + 1
            sh4.Cells(j, 3) = Int(CLng(sh2.Cells(i, 8)) - CLng(sh2.Cells(i, 7)) - 0.0001) + 1
            sh4.Cells(j, 4) = Int((CLng(sh2.Cells(i, 9)) - CLng(sh2.Cells(i, 8)) - 7) * 4 / sh2.Cells(i, 2) + 7 - 0.001) + 1
            If sh2.Cells(i, 2) > 4 Then
                sh4.Cells(j, 5) = Int((CLng(sh2.Cells(i, 10)) - CLng(sh2.Cells(i, 9))) * 8 / sh2.Cells(i, 2) - 0.001) + 1
                sh4.Cells(j, 6) = CLng(sh2.Cells(i, 11)) - CLng(sh2.Cells(i, 10))
            Else
                sh4.Cells(j, 6) = CLng(sh2.Cells(i, 11)) - CLng(sh2.Cells(i, 9))
                sh4.Cells(j, 5) = 0
            End If
            j = j + 1
        End If
    End If
Next i

NumberRows1 = sh4.Cells(sh4.Rows.count, 1).End(xlUp).Row

For b = 1 To 6
    For a = 2 To NumberRows1
    If sh4.Cells(a, b) > 0 Then
    sum = sum + sh4.Cells(a, b)
    count = count + 1
    End If
    Next a
    sh4.Cells(2, b + 7) = sum / count
    sum = 0
    count = 0
Next b


End Sub
```

Appendix G.5: KPI order in progress macro

```
Sub OrderInProgress()


Dim sh1 As Worksheet
Dim sh2 As Worksheet
Dim sh3 As Worksheet
Dim sh4 As Worksheet
Dim sh5 As Worksheet

Set sh1 = ActiveWorkbook.Sheets("Input dummy data")
Set sh2 = ActiveWorkbook.Sheets("Eventlog")
Set sh3 = ActiveWorkbook.Sheets("Capacity")
Set sh4 = ActiveWorkbook.Sheets("KPI average time per phase")
Set sh5 = ActiveWorkbook.Sheets("KPI order in progress per phase")

sh5.Cells.Clear
sh5.Cells(1, 1) = "Projects"
sh5.Cells(1, 2) = "Work preperation"
sh5.Cells(1, 3) = "Wait for release date"
sh5.Cells(1, 4) = "Warehouse"
sh5.Cells(1, 5) = "Production HTG2"
sh5.Cells(1, 6) = "Production HTG3"
sh5.Cells(1, 7) = "Shipping preperation"
sh5.Cells(1, 8) = "Orders finished"

Dim ColumnNr As Long
Dim NumberRows As Long
Dim i, j As Long
NumberRows = sh2.Cells(sh2.Rows.count, 1).End(xlUp).Row
For i = 2 To NumberRows
    ColumnNr = sh2.Cells(i, sh2.Columns.count).End(xlToLeft).Column
    If ColumnNr = 12 Then
        For j = 1 To 6
            If CLng(sh2.Cells(i, j + 5)) > 0 Then
                ColumnNr = j + 5
            End If
        Next j
    End If
    sh5.Cells(2, ColumnNr - 3) = sh5.Cells(2, ColumnNr - 3) + 1
Next i

End Sub
```

Appendix G.6: KPI warning long throughput times macro

```
Sub WarningLongThroughputTimes()

Dim sh1 As Worksheet
Dim sh2 As Worksheet
Dim sh3 As Worksheet
Dim sh4 As Worksheet
Dim sh5 As Worksheet
Dim sh6 As Worksheet
Dim sh7 As Worksheet

Set sh1 = ActiveWorkbook.Sheets("Input dummy data")
Set sh2 = ActiveWorkbook.Sheets("Eventlog")
Set sh3 = ActiveWorkbook.Sheets("Capacity")
Set sh4 = ActiveWorkbook.Sheets("KPI average time per phase")
Set sh5 = ActiveWorkbook.Sheets("KPI order in progress per phase")
Set sh6 = ActiveWorkbook.Sheets("Warning too long throughputtime")
Set sh7 = ActiveWorkbook.Sheets("Dashboard")

sh6.Cells.Clear
sh6.Cells(1, 1) = "Ordernumber"
sh6.Cells(1, 2) = "Orderclass"
sh6.Cells(1, 3) = "Projects"
sh6.Cells(1, 4) = "Work prep."
sh6.Cells(1, 5) = "Warehouse"
sh6.Cells(1, 6) = "HTG2"
sh6.Cells(1, 7) = "HTG3"
sh6.Cells(1, 8) = "Shipping prep."
sh6.Cells(1, 9) = "Total throughput time"
sh6.Cells(1, 10) = "Order finish date"


Dim a, b, c, i, j As Long
Dim NumberRows As Long

NumberRows = sh2.Cells(sh2.Rows.count, 11).End(xlUp).Row
a = 0
b = 0
c = 0
While a < 5
If CLng(sh2.Cells(NumberRows - a - b - c, 11)) > 0 Then
    If CLng(sh2.Cells(NumberRows - a - b - c, 11)) - CLng(sh2.Cells(NumberRows - a - b - c, 7)) + (CLng(sh2.Cells(NumberRows
    sh6.Cells(a + 2, 1) = sh2.Cells(NumberRows - a - b - c, 1)
    sh6.Cells(a + 2, 2) = sh2.Cells(NumberRows - a - b - c, 2)
    sh6.Cells(a + 2, 3) = CLng(sh2.Cells(NumberRows - a - b - c, 5)) - CLng(sh2.Cells(NumberRows - a - b - c, 4))
    sh6.Cells(a + 2, 4) = CLng(sh2.Cells(NumberRows - a - b - c, 6)) - CLng(sh2.Cells(NumberRows - a - b - c, 5))
    sh6.Cells(a + 2, 5) = CLng(sh2.Cells(NumberRows - a - b - c, 8)) - CLng(sh2.Cells(NumberRows - a - b - c, 7))
    sh6.Cells(a + 2, 6) = CLng(sh2.Cells(NumberRows - a - b - c, 9)) - CLng(sh2.Cells(NumberRows - a - b - c, 8))
    If sh6.Cells(a + 2, 2) > 4 Then
    sh6.Cells(a + 2, 7) = CLng(sh2.Cells(NumberRows - a - b - c, 10)) - CLng(sh2.Cells(NumberRows - a - b - c, 9))
    sh6.Cells(a + 2, 8) = CLng(sh2.Cells(NumberRows - a - b - c, 11)) - CLng(sh2.Cells(NumberRows - a - b - c, 10))
    Else
    sh6.Cells(a + 2, 8) = CLng(sh2.Cells(NumberRows - a - b - c, 11)) - CLng(sh2.Cells(NumberRows - a - b - c, 9))
    sh6.Cells(a + 2, 7) = 0
    End If
    sh6.Cells(a + 2, 9) = CLng(sh2.Cells(NumberRows - a - b - c, 11)) - CLng(sh2.Cells(NumberRows - a - b - c, 7)) + (CLng(sh2
    sh6.Cells(a + 2, 10) = CDate(sh2.Cells(NumberRows - a - b - c, 11))

    a = a + 1
    Else
    b = b + 1
    End If
Else
c = c + 1
End If
If a < 5 Then
    If b + c + a + 1 = NumberRows Then
        If a > 0 Then
        MsgBox CStr(a) + " order(s) are not completed within their throughputtime goal"
        Else
        MsgBox "All orders are completed within their throughputtime goal"
        End If
    a = 6
    End If
End If


Wend
For i = 1 To 6
    For j = 1 To 10
        sh7.Cells(i + 38, j + 5) = sh6.Cells(i, j)
    Next j
Next i
End Sub
```

Appendix G.7: KPI which department too late

```
Sub WhichdepartmentTooLate()
Dim sh1 As Worksheet
Dim sh2 As Worksheet
Dim sh3 As Worksheet
Dim sh4 As Worksheet
Dim sh5 As Worksheet
Dim sh6 As Worksheet
Dim sh7 As Worksheet

Set sh1 = ActiveWorkbook.Sheets("Input dummy data")
Set sh2 = ActiveWorkbook.Sheets("Eventlog")
Set sh3 = ActiveWorkbook.Sheets("Capacity")
Set sh4 = ActiveWorkbook.Sheets("KPI average time per phase")
Set sh5 = ActiveWorkbook.Sheets("KPI order in progress per phase")
Set sh6 = ActiveWorkbook.Sheets("Warning too long throughputtime")
Set sh7 = ActiveWorkbook.Sheets("Dashboard")
Dim a, b, c, i, j As Long

For i = 1 To 6
    For j = 1 To 10
        sh7.Cells(i + 38, j + 5) = sh6.Cells(i, j)
    Next j
Next i
For i = 1 To 5
    If sh7.Cells(i + 39, 7) = 4 Then
        If sh7.Cells(i + 39, 8) > 0.1 * 20 Then
            sh7.Cells(i + 39, 8).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 8).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 9) > 0.15 * 20 Then
            sh7.Cells(i + 39, 9).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 9).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 10) > 0.1 * 20 Then
            sh7.Cells(i + 39, 10).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 10).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 11) > 0.5 * 20 Then
            sh7.Cells(i + 39, 11).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 11).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 13) > 0.1 * 20 Then
            sh7.Cells(i + 39, 13).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 13).Interior.ColorIndex = 2
        End If
        sh7.Cells(i + 39, 12).Interior.ColorIndex = 2
    ElseIf sh7.Cells(i + 39, 7) = 8 Then
        If sh7.Cells(i + 39, 8) > 0.05 * 40 Then
            sh7.Cells(i + 39, 8).Interior.ColorIndex = 3

        Else
            sh7.Cells(i + 39, 8).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 9) > 0.125 * 40 Then
            sh7.Cells(i + 39, 9).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 9).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 10) > 0.05 * 40 Then
            sh7.Cells(i + 39, 10).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 10).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 11) > 0.375 * 40 Then
            sh7.Cells(i + 39, 11).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 11).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 12) > 0.125 * 40 Then
            sh7.Cells(i + 39, 12).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 12).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 13) > 0.05 * 40 Then
            sh7.Cells(i + 39, 13).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 13).Interior.ColorIndex = 2
        End If
    ElseIf sh7.Cells(i + 39, 7) = 12 Then
        If sh7.Cells(i + 39, 8) > 0.05 * 60 Then
            sh7.Cells(i + 39, 8).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 8).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 9) > 0.1 * 60 Then
            sh7.Cells(i + 39, 9).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 9).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 10) > 0.05 * 60 Then
            sh7.Cells(i + 39, 10).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 10).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 11) > 0.333 * 60 Then
            sh7.Cells(i + 39, 11).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 11).Interior.ColorIndex = 2
        End If
        If sh7.Cells(i + 39, 12) > 0.333 * 60 Then
            sh7.Cells(i + 39, 12).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 12).Interior.ColorIndex = 2
        End If
```

```
        If sh7.Cells(i + 39, 13) > 0.05 * 60 Then
            sh7.Cells(i + 39, 13).Interior.ColorIndex = 3
        Else
            sh7.Cells(i + 39, 13).Interior.ColorIndex = 2
        End If
    ElseIf sh7.Cells(i + 39, 5) = "" Then
        sh7.Cells(i + 39, 8).Interior.ColorIndex = 2
        sh7.Cells(i + 39, 9).Interior.ColorIndex = 2
        sh7.Cells(i + 39, 10).Interior.ColorIndex = 2
        sh7.Cells(i + 39, 11).Interior.ColorIndex = 2
        sh7.Cells(i + 39, 12).Interior.ColorIndex = 2
        sh7.Cells(i + 39, 13).Interior.ColorIndex = 2
    End If

Next i

End Sub
```

Appendix G.8: KPI planned orders in progress

```
Sub KPIDifferenceActualPlanned()

Dim sh1 As Worksheet
Dim sh2 As Worksheet
Dim sh3 As Worksheet
Dim sh4 As Worksheet
Dim sh5 As Worksheet
Dim sh6 As Worksheet
Dim sh7 As Worksheet
Dim sh8 As Worksheet
Dim sh9 As Worksheet

Set sh1 = ActiveWorkbook.Sheets("Input dummy data")
Set sh2 = ActiveWorkbook.Sheets("Eventlog")
Set sh3 = ActiveWorkbook.Sheets("Capacity")
Set sh4 = ActiveWorkbook.Sheets("KPI average time per phase")
Set sh5 = ActiveWorkbook.Sheets("KPI order in progress per phase")
Set sh6 = ActiveWorkbook.Sheets("Warning too long throughputtime")
Set sh7 = ActiveWorkbook.Sheets("Dashboard")
Set sh8 = ActiveWorkbook.Sheets("KPI planned OIP pw")
Set sh9 = ActiveWorkbook.Sheets("planning")

Dim today As Long
Dim i, j, k As Long
Dim NumberRowsPlan, NumberRowsCap, NumberRowsEvent As Long
Dim startWeek, endWeek As Long
sh8.Cells.Clear
j = 0
today = CLng(sh1.Cells(3, 3))
NumberRowsPlan = sh9.Cells(sh9.Rows.count, 1).End(xlUp).Row
NumberRowsEvent = sh2.Cells(sh2.Rows.count, 12).End(xlUp).Row
NumberRowsCap = sh3.Cells(sh3.Rows.count, 1).End(xlUp).Row
For i = 2 To NumberRowsCap
    If j > 0 Then
        j = j + 1
        sh8.Cells(j, 3) = sh3.Cells(i, 1)
        sh8.Cells(j, 2) = CDate(sh3.Cells(i, 3))
        sh8.Cells(j, 1) = CDate(sh3.Cells(i, 2))
    End If
    If today >= CLng(sh3.Cells(i, 2)) And today <= CLng(sh3.Cells(i, 3)) Then
        j = j + 1
        sh8.Cells(j + 1, 3) = sh3.Cells(i, 1)
        sh8.Cells(j, 2) = sh3.Cells(i, 3)
        sh8.Cells(j, 1) = sh3.Cells(i, 2)
    End If
Next i
sh8.Cells(1, 1) = "start"
sh8.Cells(1, 2) = "end"
sh8.Cells(1, 4) = "warehouse"
sh8.Cells(1, 5) = "HTG2"
sh8.Cells(1, 6) = "HTG3"
sh8.Cells(1, 7) = "Shipping prep"
j = 1
sh9.Cells.Clear
For i = 2 To NumberRowsEvent
     If CLng(sh2.Cells(i, 12)) >= today Then
        sh9.Cells(j, 1) = CDate(sh2.Cells(i, 12))
        sh9.Cells(j, 2) = CDate(CLng(sh2.Cells(i, 12)) + 2)
        If WorksheetFunction.Weekday(CDate(sh9.Cells(j, 2))) = 1 Then
            sh9.Cells(j, 2) = CDate(CLng(sh9.Cells(j, 2)) + 1)
        ElseIf WorksheetFunction.Weekday(CDate(sh9.Cells(j, 2))) = 7 Then
            sh9.Cells(j, 2) = CDate(CLng(sh9.Cells(j, 2)) + 2)
        End If
        If sh2.Cells(i, 2) = 4 Then                                          'days for assembly
            sh9.Cells(j, 3) = CDate(CLng(sh9.Cells(j, 2)) + 14)
        ElseIf sh2.Cells(i, 2) = 8 Then
            sh9.Cells(j, 3) = CDate(CLng(sh9.Cells(j, 2)) + 21)
        ElseIf sh2.Cells(i, 2) = 12 Then
            sh9.Cells(j, 3) = CDate(CLng(sh9.Cells(j, 2)) + 28)
        End If
        If WorksheetFunction.Weekday(CDate(sh9.Cells(j, 3))) = 1 Then
            sh9.Cells(j, 3) = CDate(CLng(sh9.Cells(j, 3)) + 1)
        ElseIf WorksheetFunction.Weekday(CDate(sh9.Cells(j, 3))) = 7 Then
            sh9.Cells(j, 3) = CDate(CLng(sh9.Cells(j, 3)) + 2)
        End If
        If sh2.Cells(i, 2) = 4 Then                                          'days for assembly
            sh9.Cells(j, 4) = 0
        ElseIf sh2.Cells(i, 2) = 8 Then
            sh9.Cells(j, 4) = CDate(CLng(sh9.Cells(j, 3)) + 7)
        ElseIf sh2.Cells(i, 2) = 12 Then
            sh9.Cells(j, 4) = CDate(CLng(sh9.Cells(j, 3)) + 28)
        End If
        If WorksheetFunction.Weekday(CDate(sh9.Cells(j, 4))) = 1 Then
            sh9.Cells(j, 4) = CDate(CLng(sh9.Cells(j, 4)) + 1)
        ElseIf WorksheetFunction.Weekday(CDate(sh9.Cells(j, 4))) = 7 Then
            sh9.Cells(j, 4) = CDate(CLng(sh9.Cells(j, 4)) + 2)
        End If
        sh9.Cells(j, 5) = CDate(CLng(sh9.Cells(j, 4)) + 3)
        If WorksheetFunction.Weekday(CDate(sh9.Cells(j, 5))) = 1 Then
            sh9.Cells(j, 5) = CDate(CLng(sh9.Cells(j, 5)) + 1)
        ElseIf WorksheetFunction.Weekday(CDate(sh9.Cells(j, 5))) = 7 Then
            sh9.Cells(j, 5) = CDate(CLng(sh9.Cells(j, 5)) + 2)
        End If
        j = j + 1
    End If
Next i

For j = 2 To 11
    For i = 2 To NumberRowsPlan
        If (CLng(sh8.Cells(j, 1)) >= CLng(sh9.Cells(i, 1)) And CLng(sh8.Cells(j, 1)) <= CLng(sh9.Cells(i, 2))) Or (CLng(sh8.
            sh8.Cells(j, 4) = sh8.Cells(j, 4) + 1
        ElseIf CLng(sh8.Cells(j, 1)) <= CLng(sh9.Cells(i, 1)) And CLng(sh8.Cells(j, 2)) >= CLng(sh9.Cells(i, 2)) Then
            sh8.Cells(j, 4) = sh8.Cells(j, 4) + 1
        End If
        If (CLng(sh8.Cells(j, 2)) >= CLng(sh9.Cells(i, 2)) And CLng(sh8.Cells(j, 2)) <= CLng(sh9.Cells(i, 3))) Or (CLng(sh8.
            sh8.Cells(j, 5) = sh8.Cells(j, 5) + 1
```

```
        ElseIf CLng(sh8.Cells(j, 2)) <= CLng(sh9.Cells(i, 2)) And CLng(sh8.Cells(j, 3)) >= CLng(sh9.Cells(i, 3)) Then
            sh8.Cells(j, 5) = sh8.Cells(j, 5) + 1
        End If
        If (CLng(sh8.Cells(j, 3)) >= CLng(sh9.Cells(i, 3)) And CLng(sh8.Cells(j, 3)) <= CLng(sh9.Cells(i, 4))) Or (CLng(sh8.
            sh8.Cells(j, 6) = sh8.Cells(j, 6) + 1
        ElseIf CLng(sh8.Cells(j, 3)) <= CLng(sh9.Cells(i, 3)) And CLng(sh8.Cells(j, 4)) >= CLng(sh9.Cells(i, 4)) Then
            sh8.Cells(j, 6) = sh8.Cells(j, 6) + 1
        End If
        If (CLng(sh8.Cells(j, 4)) >= CLng(sh9.Cells(i, 4)) And CLng(sh8.Cells(j, 4)) <= CLng(sh9.Cells(i, 5))) Or (CLng(sh8.
            sh8.Cells(j, 7) = sh8.Cells(j, 7) + 1
        ElseIf CLng(sh8.Cells(j, 4)) <= CLng(sh9.Cells(i, 4)) And CLng(sh8.Cells(j, 5)) >= CLng(sh9.Cells(i, 5)) Then
            sh8.Cells(j, 7) = sh8.Cells(j, 7) + 1
        End If
    Next i
Next j


End Sub
```

Appendix G.9: KPI number of FTEs

```
Sub numberFTEs()
Dim sh1 As Worksheet
Dim sh2 As Worksheet
Dim sh3 As Worksheet
Dim sh4 As Worksheet
Dim sh5 As Worksheet
Dim sh6 As Worksheet
Dim sh7 As Worksheet
Dim sh8 As Worksheet
Dim sh9 As Worksheet

Set sh1 = ActiveWorkbook.Sheets("Input dummy data")
Set sh2 = ActiveWorkbook.Sheets("Eventlog")
Set sh3 = ActiveWorkbook.Sheets("Capacity")
Set sh4 = ActiveWorkbook.Sheets("KPI average time per phase")
Set sh5 = ActiveWorkbook.Sheets("KPI order in progress per phase")
Set sh6 = ActiveWorkbook.Sheets("Warning too long throughputtime")
Set sh7 = ActiveWorkbook.Sheets("Dashboard")
Set sh8 = ActiveWorkbook.Sheets("KPI planned OIP pw")
Set sh9 = ActiveWorkbook.Sheets("planning")

Dim today As Long
Dim i, j, k As Long
Dim NumberRowsPlan, NumberRowsCap, NumberRowsEvent As Long
Dim startWeek, endWeek As Long
today = CLng(sh1.Cells(3, 3))
NumberRowsPlan = sh9.Cells(sh9.Rows.count, 1).End(xlUp).Row
NumberRowsEvent = sh2.Cells(sh2.Rows.count, 12).End(xlUp).Row
NumberRowsCap = sh3.Cells(sh3.Rows.count, 1).End(xlUp).Row
j = 1
sh8.Cells(14, 2) = "warehouse"
sh8.Cells(14, 3) = "HTG2"
sh8.Cells(14, 4) = "HTG3"
sh8.Cells(14, 5) = "Shipping prep"

For i = 2 To NumberRowsCap
    If sh3.Cells(i, 1) >= sh8.Cells(2, 3) Then

        sh8.Cells(j + 14, 1) = sh3.Cells(i, 1)
        sh8.Cells(j + 14, 2) = sh3.Cells(i, 6)
        sh8.Cells(j + 14, 3) = sh3.Cells(i, 7)
        sh8.Cells(j + 14, 4) = sh3.Cells(i, 8)
        sh8.Cells(j + 14, 5) = sh3.Cells(i, 9)
        j = j + 1
    ElseIf j > 1 And j < 11 Then
        sh8.Cells(j + 14, 1) = sh3.Cells(i, 1)
        sh8.Cells(j + 14, 2) = sh3.Cells(i, 6)
        sh8.Cells(j + 14, 3) = sh3.Cells(i, 7)
        sh8.Cells(j + 14, 4) = sh3.Cells(i, 8)
        sh8.Cells(j + 14, 5) = sh3.Cells(i, 9)
    End If
Next i
End Sub
```