Master Thesis

The Automated Generation of User Specific Dashboards in a Multi-tenant ERP Product

Jasper van de Wiel

Master Business Informatics
Dept. of Information and Computing Science
Utrecht University

Utrecht University

Author	Jasper van de Wiel
	J.M.J.vandewiel@students.uu.nl
	Master Business Informatics
	Utrecht University
First Supervisor	dr. Slinger Jansen
	Slinger.jansen@uu.nl
	Department of Information and Computing Sciences
	Utrecht University
Second Supervisor	dr. Marco Spruit
	M.R.Spruit@uu.nl
	Department of Information and Computing Sciences
	Utrecht University
External Supervisor	Ruben Mijwaart
	Software Product Manager
	Exact Software Nederland B.V.
Eternal Supervisor	Jeroen Gruppen
	Software Engineering Manager
	Exact Software Nederland B.V.

I. Abstract

Users of ERP software are typically presented with a one size fits all product, even though each user has his own way of working. Customization of functionality by software producing organizations is a complex manual task. In an attempt to find a solution for this problem this research is focused on linking usage patterns to users of SaaS ERP systems, based on their ERP system usage data. These usage patterns are used in order to automatically provide the users with business intelligence dashboards, which are tailored to their specific information needs. In order to find a way to discover different usage patterns based on usage data the following research question has been formulated: “How can a model of user information needs be developed from ERP usage statistics in order to generate user specific BI dashboards?”

This research contributes to the body of scientific knowledge on the subject of personalized BI dashboards by providing a method. This method can be used in order to develop a model that provides a systematic way to discover different usage patterns based on the usage data of an ERP system. This model has been created by performing web usage mining on a big dataset containing over 15.5 million user actions in a SaaS ERP product. In order to find a method for creating a model of usage patterns a number of techniques including process mining and association rule mining have been studied. The final method used to perform web usage mining is content enhanced clustering. The clustering technique used is Formal Concept Analysis. The data consists of usage data, semantic data and usage data of the ERP system.

The model has been evaluated with end-users of a large SaaS ERP software product by using a prototype of a dashboard generator making use of the developed model in an extensive case study. The results of the case study show that the generated dashboards are useful, accurate, and save the end user time in composing their own. The research process and the findings of this research are documented in this document.

II. Acknowledgements

This document is the result of hours of hard work, consisting of research on the application of usage data in order to provide a personalized experience to users of ERP systems. The research process that is the basis for this thesis has been a great educational experience for me. I have performed literature research on a variety of interesting topics, I have had the chance to experiment with multiple datamining techniques, I have spoken to many experts in the field of ERP systems, developed a prototype, and evaluated this prototype with end-users of an ERP system.

I would like to express my gratitude to my first supervisor dr. Slinger Jansen for his motivating and constructive criticism during our meetings and digital conversations. I would like to thank dr. Marco Spruit for his feedback on this thesis, and his sharp eye. Thanks also to my external supervisor Ruben Mijwaart, former product manager at Exact, for his guidance on the direction of my research. Of course, a big thank you to all other people at Exact that made it possible for me to perform my research at an innovative software company. A special thanks to my colleagues of the connectivity team that I have had the pleasure of working with for the past year.

Last but not least, I want to thank my friends and family, who have all supported me during my research. A special thanks to Dieuwertje, for supporting me morally, and more importantly for bearing with me through this time-consuming period.

Have a good read!

Jasper van de Wiel
Tilburg, January 2017.

III. Table of contents

I. Abstract ..... **V**

II. Acknowledgements ..... **VII**

III. Table of contents ..... **IX**

1. Introduction **1**

1.1 Background 1

1.2 Research Focus 1

1.3 Overall Research Aim and Individual Research Objectives 2

2. Research Method **4**

2.1 Research Overview 4

2.2 Design Science 6

2.3 Literature Review Protocol 8

2.4 Model Development 8

2.5 Evaluation 9

3. Theoretical Background **11**

3.1 Software as a Service 11

3.2 ERP as a Service 13

3.3 Multi-Tenancy 13

3.4 Business Intelligence 14

3.5 Customization & Configuration 15

3.6 Web Usage Mining 16

4. Model Development - Data Gathering and Pre-processing **23**

4.1 Data Gathering 23

4.2 Data Pre-Processing 25

5. Model Development - Methods Considered for Pattern Discovery **28**

5.1 Association Rule Mining 28

5.2 Process mining 29

5.3 Formal Concept Analysis 30

5.4 Chapter Conclusion 33

6. Model Development – Pattern Discovery **34**

6.1 The Dataset 34

6.2 Performing Content Enhanced FCA 35

6.3 Chapter Conclusion 40

7. Model Development – Pattern Analysis 41

7.1 Pattern Analysis 41

7.2 Expanding the model 43

7.3 Chapter Conclusion 45

8. Prototype 46

9. Evaluation 50

9.1 Case Study Protocol 50

9.2 Case Study Participants 50

10. Results 51

10.1 Case Study Results per Package 51

10.2 Chapter Conclusion 53

11. Discussion & Future Work 54

11.1 Discussion 54

11.2 Future work 55

12. Conclusion 56

References 59

Appendices 63

Appendix A. Aspx'es, Related (Sub)Menu Items, and Title Elements 63

Appendix B. Deleted Aspx'es 68

Appendix C. Case Study Results 69

1. Introduction

1.1 Background

More than ten years ago, Kaplan (2005) observed that a great number of software companies were trying to make a switch from on premise deployment towards the Software as a Service business model. The SaaS model is especially interesting for Small and Medium businesses (SMEs) because these enterprises usually have a small IT budget (Bezemer, Zaidman, Platzbeecker, Hurkmans, Hart, 2010; Wang, Guo, Gao, Sun, Zhang, An, 2008). Gartner uses the number of employees and a business's revenue as a measure to classify a business as an SME. Small businesses are businesses with less than 100 employees, and midsize companies have a max of 999 employees. For revenue, the boundaries lay with respectively $50 million and $1 billion in annual revenue. SMEs have a strong need for business software, for instance Enterprise Resource Planning (ERP) systems. In particular business software as a service is interesting to SMEs because the on premise deployment of for instance an ERP system comes at a high price (Makkar & Bist, 2012).

A popular architectural pattern used by ERP providers for delivering software as a service is multi-tenancy. With the use of the multi-tenancy model, only one instance of the software is running on the side of the software provider. All the customers (tenants) of the software provider access the same instance of the software (Bezemer & Zaidman, 2010). A tenant usually does not refer to one person, but rather to (part of) an organization. Each tenant has its own specific requirements to the software (Kabbedijk & Jansen, 2011). Multi-tenancy enables the software provider to benefit fully from the economy of scale by making serving multiple clients more scalable and cost-effective.

One key feature of ERP systems is providing its users with reports on business information. This is realized by implementing some form of Business intelligence (Later referred to as BI) functionality. In ERP systems, BI is often put into practice in the form of dashboards. Dashboards are data visualization tools that provide visual displays of important information, preferably on a single screen so that the information can be processed and explored quickly (Palpanas, Chowdhary, Mihaila & Pinel, 2007). A dashboard can for instance help to visualize a trend in the data that is hard to see from the data itself by showing the data in a graph. Dashboards play a big role in today's business environments. According to Tutunea and Rus (2012), dashboards have gained ground in recent years due to the lowering costs of implementing BI.

1.2 Research Focus

Despite the large adaptation of dashboards in businesses the design and implementation of dashboards is still prone to failure (Aldridge, 2008; Few, 2006). A big cause of the failure of dashboards is the dashboard not showing the right information (Few, 2006).

Aligning dashboards with a company's business goals and characteristics is challenging

(Aldridge, 2008; Eckerson, 2003; Kawamoto & Mathers, 2007; Watson et al., 2006, 2007).

To help a user grasp what is on the dashboard the information should be linked to relevant concepts in the business context (Frank, Heise & Kattenstroth, 2009). This misalignment is often caused by providing a dashboard that is not tailored to the specific business characteristics of a company. If the BI solution does not match with the user's characteristics and goals, it will just be a cool gimmick instead of a useful tool providing its users with relevant information about their companies' performance. This in turn leads to users looking for their information in other places (Eckerson, 2010).

In accordance with dashboards, ERP systems also need to be tailored to the needs of its users (Kumar & Hillegersberg, 2000). On top of that, multi-tenant systems should also be tailored to the needs of its users. Multi-tenant systems need a high degree of tailorization (Bezemer et al., 2010b). The need for tailoring in multi-tenant systems follows from the difference between the multi-user architecture and the multi-tenant architecture (explained in section 1.1). The single software instance hosted by the software provider should be tailored to the requirements of the different tenants using this single instance (Kwok, Nguyen, and Lam, 2008).

It is evident that for ERP systems, multi-tenant software, and dashboards it is important that the content provided to the user is tailored to the needs of the specific user. This means that a high degree of tailorization is certainly needed for tenant-specific reporting in multi-tenant ERP systems. Tailoring is a way of providing flexibility to the user. A problem with providing users with more flexibility in the software is that the usability of the system will decrease. This is the so-called flexibility-usability tradeoff (Steinebach, Guhathakurta & Hagen, 2009). The reason for the decrease in usability is the increasing number of options a user is confronted with in the user interface of a highly configurable software product. The background information in the previous and current section leads to the following problem statement:

SaaS ERP vendors do not have valid approaches to customize BI dashboards to the needs of individual users.

1.3 Overall Research Aim and Individual Research Objectives

If the process of customizing to the specific needs of a user can be automated in such a way that the user is provided with user specific reports, there is no need for flexibility on the user side. This means the usability problem is not relevant anymore. On top of that, the chance of failure of the reporting tool is also decreased because users will always get relevant reports. The overall aim of this research derived from above statements is: To find a way to automatically generate dashboards tailored to the information needs of individual employees of SMEs in a multi-tenant ERP software product. This overall research aim has been transformed into the following main research question:

RQ: How can a model of user information needs be developed from ERP usage statistics in order to generate user specific BI dashboards?

In order to structure the research, the main research question has been split into four sub research questions. The combination of answers to these sub questions answers the main research question.

SRQ1: How can the different usage patterns of SME employees while using ERP systems be identified?

This question is the first question that needs to be answered. In order to answer this question a number of different data mining methods that might be useful in order to identify different usage patterns have been studied. The next goal is to identify the different usage patterns of SME employees while using ERP systems. This leads to the second research question:

SRQ2: What are the different usage patterns of SME employees while using ERP systems?

If different users of ERP systems have different usage patterns, the next goal is to find out what the specific information needs of the different users expressed by the usage patterns are. This is the basis for the third research question.

SRQ3: How do the usage patterns express different information needs of SME employees while using ERP systems?

The formulation of this third research question suggests there is certainty that the information needs of employees of SMEs using ERP systems differ from each other. However, the answer to this research question might as well be they do not differ. This means there is a need to find out if employees of SMEs making use of ERP systems have different information needs, and if so, how they are different from each other. If the different users all have the same information needs, there is no need to generate dashboards configured to the specific information needs of each tenant. Because if every user has the same information needs a general dashboard is sufficient in covering the needs of all the users.

SRQ4: To what extent does a dashboard, generated based on usage patterns, satisfy the information needs of individual SME employees?

If the usage patterns and related information needs of ERP software users can be identified, a dashboard generation tool that provides each user with a dashboard, automatically generated based on the information needs of that user can be developed. When this tool has been developed, the dashboards generated by this tool can be evaluated in order to see if the information needs as identified by the generator are truly the information needs the user has.

A short overview of what research methods are used and how they are linked to the different research questions, is depicted in the research model in figure 1. The research methods are explained in more detail in the next chapter.

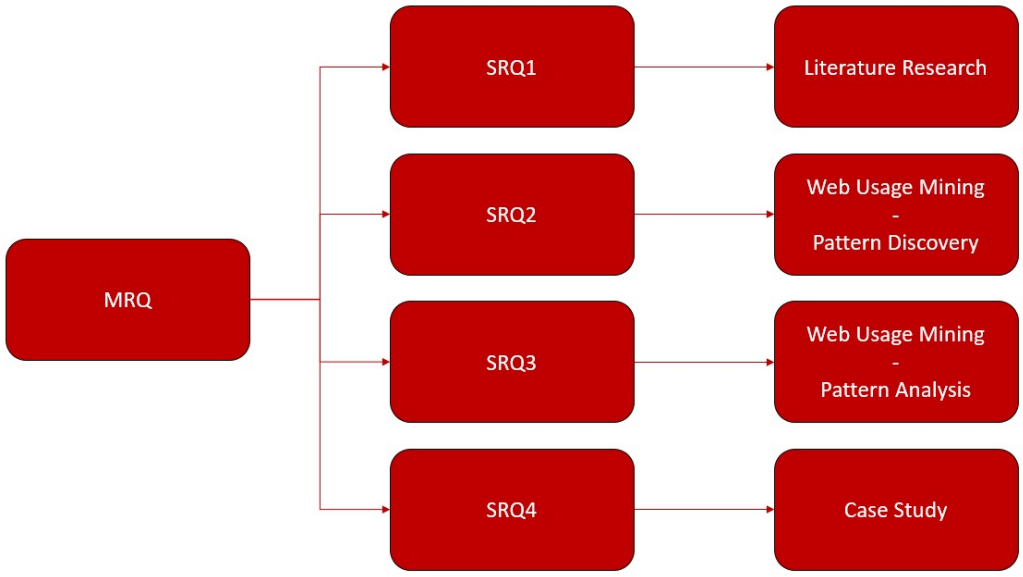

Figure 1. Research model

2. Research Method

2.1 Research Overview

In order to answer the research questions stated in section 1.3 the research has been conducted in four consecutive steps as depicted in the Process-Deliverable Diagram (PDD) in figure 2. This PDD shows a high-level overview of the research. A number of different methods are used throughout this research in order to answer the research questions in a scientifically sound manner. However, all steps are conducted by using an overarching research paradigm called Design Science. The design science paradigm has been employed using the framework as described by Peffers Tuunanen, Rothenberger and Chatterjee (2007), making use of the guidelines as proposed by Hevner, March, Park and Ram (2004), and taking into account the refinements to these guidelines as described by Wieringa (2009). The design science paradigm is suitable for this research because an organizational problem has been identified that needs to be solved. The need for the automatic generation of business intelligence dashboards tailored to the information needs of individual tenants making use of ERP software products has been identified. This need has been identified from scientific literature by performing a literature review, and is supported by a business need. An artefact has been developed, that contributes to solving the organizational problem. In addition to this, the research helps answering the main research question and thus, adds valuable findings to the current body of knowledge. These two reasons match exactly the criteria for using the Design Science paradigm.

Two different methods are used in order to gather the data for this research. One method is the case study research method. The other method is a data mining approach called web usage mining. The case study research method is used at different steps in the research and is applied by using a number of different techniques. For example, client interviews and expert interviews have been used in order to evaluate the model that follows from applying the data mining techniques, but also prototype development and thinking aloud sessions have been used in order to evaluate the deliverables created in this research. The company that acts as the case study company in this research provides the data used in the data mining tasks.

The design science paradigm, and the way in which it has been applied in this research, is explained in more detail in section 2.2. In section 2.3, the literature review protocol is presented. Section 2.4 elaborates on the ontology creation and section 2.5 is concerned with the evaluation techniques used during this research.

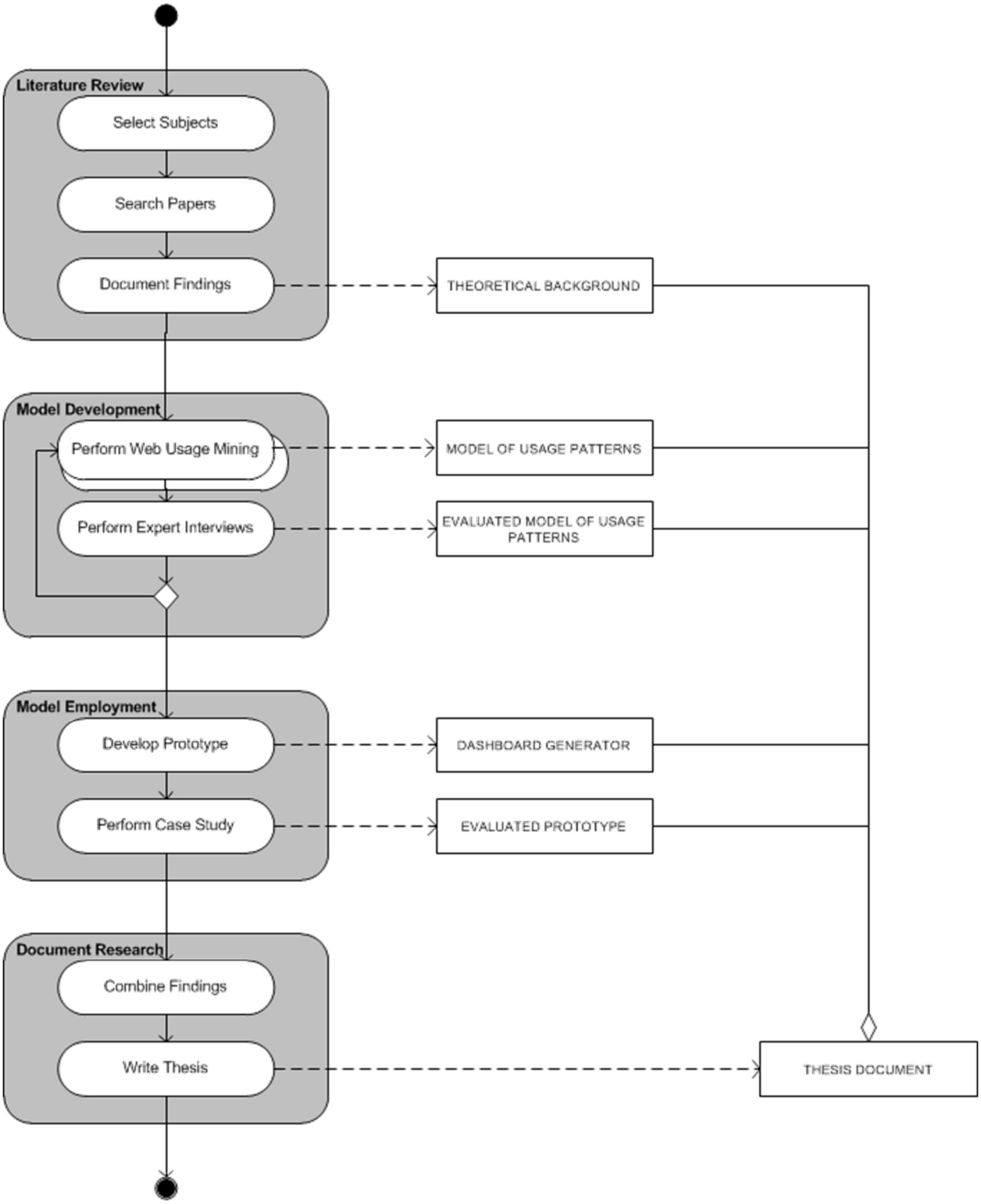

Figure 2. Research overview

2.2 Design Science

In order to execute the research, and answer the research questions, the design science paradigm has been applied. There are a number of ways of applying the design science paradigm, this research has been conducted using the design science research methodology (DSRM) developed by Peffers et al. (2007). This methodology is based on the design science research process model (DSRP) as defined by Peffers, Tuunanen, Gengler, Rossi, Huis, Virtanen and Bragge (2006). In addition, in order to conduct solid design science research the seven guidelines described by Hevner et al. (2004) are adhered to, taking into account the proposed refinements by Wieringa (2009).

In order to give shape to the design science research process the DS process model has been used, which is part of the DSRM as proposed by Peffers et al. (2007). The authors have developed a process for design science research for information systems that is consistent with earlier keystone literature on DS research. The process provides a nominal process model for performing design science research, and it provides a mental model for presenting and appreciating design science research for information systems. The consistency with earlier literature is accomplished by looking at leading literature on design science research. Seven representative DS papers (Archer, 1984; Eekels & Roozenburg, 1991; Hevner et al., 2004; Nunamaker, Chen, Purdin, 1991; Rosi & Sein, 2003; Takeda, Veerkamp, Tomiyama & Yoshikawam, 1990; Walls, Widmeyer & Sawy, 1992) have been compared and Peffers et al. conclude that the authors agree substantially on common elements. This results in the DS process model consisting of six steps in a nominal sequence. The model has successfully been applied by the authors themselves for at least four times, and has been applied by other researchers numerous times. The application of the method is considered successful if the paper describing the research has been published. The fact that the model has been used in successful research proves that the process model is nominal and that it provides a mental model for presenting design science research for information systems.

The process consists of the following six steps:

1. Problem identification and motivation;
2. Objectives of a solution;
3. Design and development;
4. Demonstration;
5. Evaluation;
6. Communication.

An adaptation of the original visualization of the DS process model by Peffers et al., applied to our research approach, is depicted in figure 3. In this research the problem centered approach has been followed by starting at the “Identify Problem & Motivate” step in the model. The problem has been identified and motivated by looking into scientific literature, this has been done briefly in the introduction of this thesis and in more detail in the literature review. The objectives of the solution are partly specified based on literature; the other part is specified based on expert interviews. The design and development step is brought into practice by developing a model making use of data mining. This model is demonstrated by putting it into practice in a prototype, which provides dashboards based on the model. This prototype has been put to practice in a case study. The model is evaluated by comparing the objectives of the prototype to the actual observed results from the use of the prototype in the case study. Finally, the communication step is completed by writing down our findings in this document.

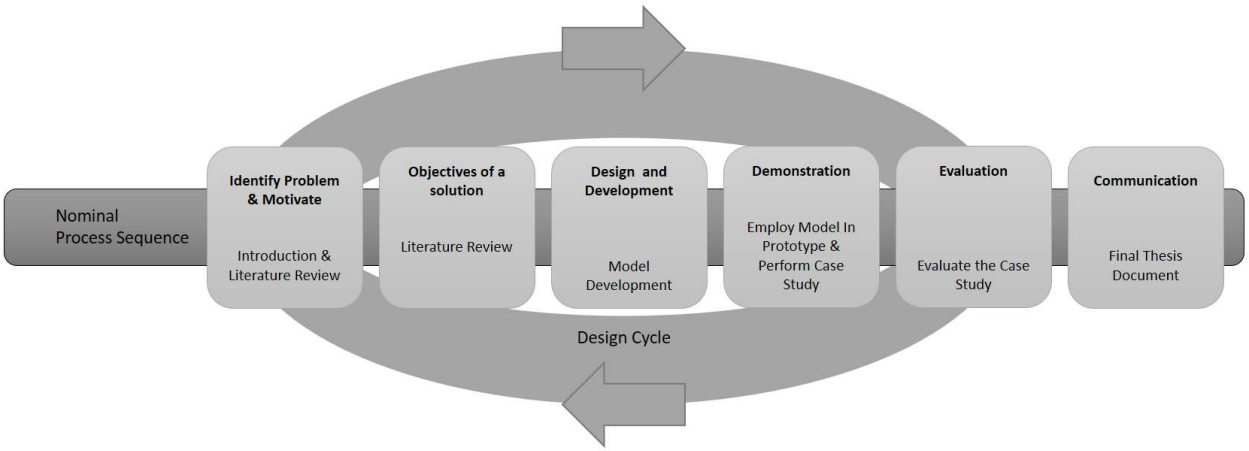

Figure 3. Adaptation of the DS process model

In addition to the process sequence by Peffers et al. (2007) the seven design science guidelines as proposed by Hevner et al. (2004) are adhered to, taking into account the refinements proposed by Wieringa (2009). In table 1 below, the design science approach of this research has been mapped out to the steps of the DS process model. In the right most column references to the guidelines of Hevner et al. (2004) have been placed at the steps these guidelines are applied to.

Table 1. DS process model mapped to activities and guidelines.

DS process model step	**Activities**	**Guideline (Hevner et al., 2004)**
1. Problem identification & motivation	This research started by identifying and **defining a problem statement** in chapter 1. The importance and need of this research has been motivated in chapter 1 as well. This has been done to a larger extent in the **literature review** documented in chapter 3.	Problem Relevance
2. Objectives of a solution	The objectives of the solution are also gathered from **literature**.	Problem Relevance
3. Design and development	The design and development of the artifact (**The Model**) is described in chapter 4, 5, 6 and 7.	Design as an artifact, Research Rigor, Design as a Search Process
4. Demonstration	The demonstration has been completed by bringing the model into practice in a **prototype** which provides users with user specific dashboards created based on the model. The prototype is described in chapter 8.	Design Evaluation, Design as a search process
5. Evaluation	The evaluation is performed in the form of a **case study** in which users of the SaaS ERP system use the developed artefact. This case study is described in chapter 9, and the results are described in chapter 10.	Design Evaluation, Research Rigor, Design as a search Process
6. Communication	The final **thesis document** is the way in which this research is communicated.	Research contribution, Communication of research

2.3 Literature Review Protocol

The first step of this research has been completed by performing a literature review. The literature review provides the necessary background information in order to answer the research questions in the next steps of the research. The fields of study of this research are 'Business Intelligence', 'Dashboards', 'ERP as a Service', 'Formal concept analysis', 'Multi Tenancy', 'Software as a Service', 'Tailoring', and 'Web usage mining'. A quick overview of how these subjects are related to the research proposed in this document has been provided in sections 1.1 and 1.2. In the literature review, each topic is discussed in more detail. A topic that is mentioned in the introduction, but is part of this literature review, is the topic of web usage mining.

In the literature review google scholar and the online computer science bibliography DBLP are used in order to gather relevant papers on the subjects of research. The first set of papers has been gathered based on (equivalents and/or synonyms of) the following search terms and combinations thereof: 'Software as a Service', 'Multi-tenant software', 'Business intelligence', 'Dashboards', 'Tailoring' and 'Web Usage Mining'. Some examples of what is meant with synonyms/equivalents are: 'SaaS' instead of 'Software as a Service' and 'BI' instead of 'Business Intelligence'. But also: 'Configured' instead of 'Tailored' and 'Reporting' instead of 'Dashboards'. Some examples of what is meant with "combinations thereof" are: 'Tailored dashboards', 'Configurable Business Intelligence' and 'BI in Multi-tenant software'.

More papers have been gathered by applying the forward and backward reference search method on the first set of papers. The backward reference searching method involves looking into the references that are cited in the first set of papers, in order to find additional interesting papers. The forward reference searching method is concerned with looking for papers that have cited one of the papers that are selected in the first set of papers. This means looking into publications created after the initial article has been created. After completing the search for papers based on the forward and backward reference search method a first selection of papers is made based on the titles of the papers. This lead to a smaller set of papers out of which a relevant subset has been selected by reading the abstracts. The last selection is based on the full text of the papers, after the last selection round, a set of papers remains that consists of papers that are relevant to the scope of this project. This set of papers has been used to write a thorough literature review on which the rest of this research project is based.

2.4 Model Development

In order to discover the different usage patterns of the users, web usage mining has been used. The first step performed in order to apply web usage mining is data gathering. The data has been gathered by querying the activity log of a big SaaS ERP product, and by scraping data from the pages of this ERP system. Next pattern discovery has been performed by applying formal concept analysis on the gathered data. Because usage data is combined with content and structure data, this method is called content-enhanced formal context analysis. The application of content-enhanced formal context analysis and pattern analysis results in different usage patterns based on the ERP usage of different users.

2.5 Evaluation

In the evaluation of the model of different users and their information needs, the first step is to perform expert interviews with experts in the domain of ERP systems. After the model has been evaluated with experts, a prototype of a dashboard generator has been developed. This prototype has been used in a case study in order to evaluate that the model is not only correct but that it can also be applied in order to generate meaningful dashboards for users of an ERP system. This has been validated by checking if the dashboards generated based on the model deliver valuable and relevant information to the users within the different usage patterns.

2.5.1 Expert Interviews

The initial model is based on findings in the data. However, in order to evaluate the model with expert knowledge, experts in the field have been interviewed. In order to do this, expert interviews have been conducted with software product marketeers, product managers, employees of a customer intelligence team and software developers. The experts all have experience working at a big SaaS ERP provider.

2.5.2 Prototype

The prototype has been developed using PHP. The prototype has been developed as an app for the app center of a big SaaS ERP product. The application pulls the app users' data from the SaaS ERP product via a REST API. This data is used in order to compare the users' usage data to the model, and link the user to a usage pattern in the model. The usage pattern the user is linked to prescribes what components should be shown in the dashboard.

The dashboard is generated by pulling the prescribed data from the user data by, again, making use of the REST API. And pushing the data into Microsoft power BI. The dashboard that is formed by doing this is shown to the user in the web app. Microsoft PowerBI has been used because it provides a quick and easy way to create an appealing and clear dashboard, and therefore is a suitable tool for prototyping.

2.5.3 Case study

The case study protocol is constructed by following the guidelines provided by Yin (2009). According to Yin (2009) there are three types of case studies: descriptive, exploratory, and explanatory case studies. Moreover, there are four kinds of case study designs: Single-case holistic designs, single-case embedded designs, multiple-case holistic designs, and multiple-case embedded designs. These four kinds of case study designs are depicted in figure 4 below. The distinction between the single- and multiple-case design depends, as the name suggests, on if a single case or multiple cases are being studied whilst conducting the case study.

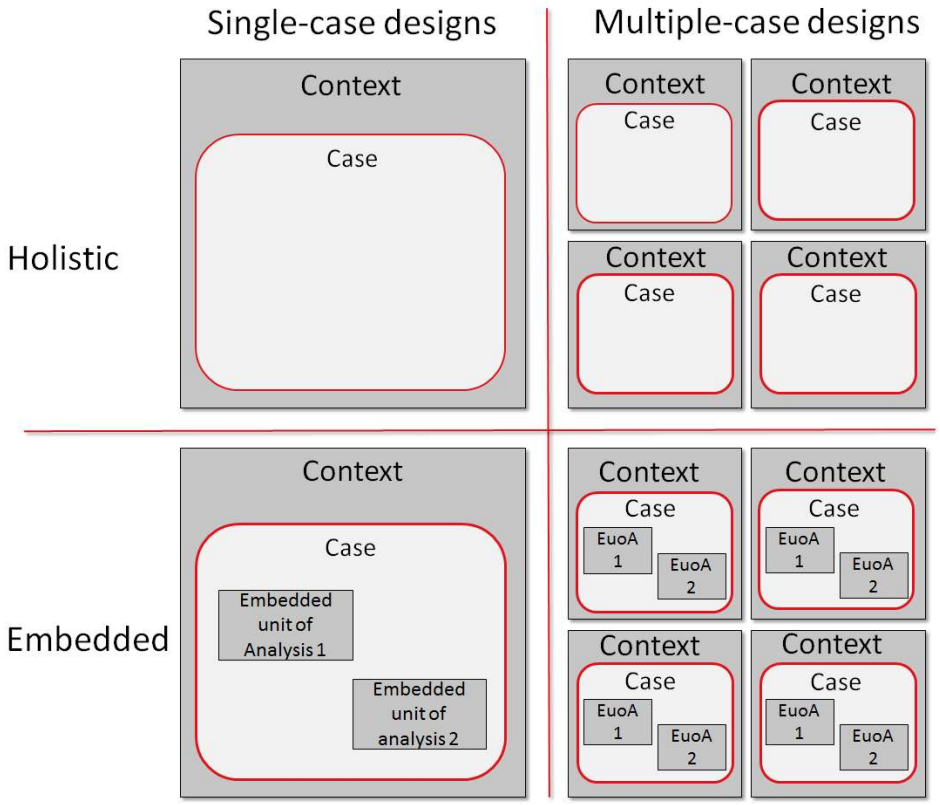

Figure 4. Case study designs (Yin, 2009)

The case study in this research is a single case embedded case study. A single case with multiple units of analysis has been studied. The case study is both exploratory and explanatory. The case study has been executed by evaluating the created prototype with a number of customers of a SaaS ERP product. The prototype has been evaluated by first providing the case study subjects with all possible dashboard components showing dummy data. The users are told to pick the components they would like to see in their personal dashboard. Next, a dashboard is shown that, according to the developed model, fits the information needs of this specific user. This dashboard contains the real user data. The components in this dashboard have been compared to the ones picked by the user in the first test. If there are any differences the user is asked if they still support their initial opinion, or if they like the generated dashboard better. The differences have been discussed thoroughly in order to check if the generated dashboard satisfies the user's needs.

3. Theoretical Background

In this chapter, the major topics involved in this research are discussed. The information for this chapter has been collected by executing the literature review as described in section 2.3. The topics covered in this literature review are software as a service, ERP as a Service, Multi-Tenancy, Business Intelligence, tailoring and web usage mining. The literature available on each of these topics has been reviewed in a separate section.

3.1 Software as a Service

SaaS is software that, contrary to on premise software, is accessed by the user via the internet (Dubey and Wagle, 2007). The on premise business model involves the deployment and use of software at the physical location of the customer. Whereas with the SaaS business model the software is hosted by the software vendor and the user accesses the software via the internet. So instead of buying a packaged software product the software is leased by the customer. The customer pays the software provider by for example a 'per-user' or 'per-month' subscription fee (Kaplan, 2007). As stated in the introduction, for the past ten years numerous software companies have been making a switch from on premise software to the SaaS business model. This switch has especially been focused on the enterprise software domain (Kabbedijk, Bezemer, Jansen, Zaidman, 2014). This switch became possible because of the advancing of technology and the rise of the web as the main medium for communication (Nitu, 2009). Customers were getting more used to the web and therefore became more comfortable doing business via the internet (Mertz, Eschinger, Eid & Pring, 2007).

Software vendors identified a possible gold mine by providing SMEs with a small IT budget with software via the SaaS model. SMEs often could not afford expensive on-premises or ASP software (Bezemer et al., 2010a; Wang et al., 2008). On premise software requires a big one-time investment at the moment of purchase and is expensive to maintain (D'souza et al., 2012). And ASP software was costly to maintain and set-up for software vendors, these costs were piped to the customers of ASP software (Nitu, 2009). SaaS applications do not require a big investment since the consumer usually pays a monthly fee (Kaplan, 2007), this is why SaaS is gaining this much popularity on the consumer side. Moreover, SaaS applications are cheaper to maintain for the provider, since there is only a single code base to maintain. This means lower costs for the provider and therefore lower costs for the customer. Another aspect which saves the customer money is the fact that SaaS applications are centrally managed at the side of the software provider, this means that the customer does not need to hire in-house experts on the software (Kaplan, 2007; Nitu, 2009; Wang, Laszewski, Younge, He, 2010). For software providers the decreased time to market is one of the main reasons why SaaS is gaining so much popularity (Aulbach, Grust, Jacobs, Kemper & Rittinger, 2008).

Where the benefits of SaaS are to a large extent agreed upon by the scientific community, there is no consensus on the definition of SaaS yet. Numerous definitions for SaaS have been provided in literature. Mäkilä, Järvi, Rönkkö and Nissilä (2010) capture eight of these definitions, and have identified "five distinct characteristics that are typically associated with SaaS". These five distinct characteristics are the following:

1. Product is used through a web browser;
2. Product is not tailor made for each customer;
3. The product does not include software that needs to be installed at the customer's location;
4. The product does not require special integration and installation work;
5. The pricing of the product is based on actual usage of the software (Mäkilä et al., 2010).

These characteristics are important characteristics of SaaS; however, the authors do not provide a unified definition for SaaS. Moreover, in our opinion the authors in their selection of papers defining SaaS missed a few important ones. The most important paper they missed is the paper by Chong and Carraro (2006). They give the following definition of SaaS: "Software deployed as a hosted service and accessed over the Internet."

As mentioned by the authors themselves this definition is a little simplistic, since it also describes simple, non-mature, SaaS solutions. However, the definition is in accordance with other authors' views on SaaS. For example, Buxmann, Hess and Lehmann (2008) describe SaaS as providing a software solution to customers as a service via the internet, in which the provider of the software hosts the software. And Dubey and Wagle (2007) write about delivering services that can be accessed and used via the internet. Thus, what is missing in the simple definition of Chong et al. is the notion that the provider and the user are different parties. Chong et al. complement their definition by providing a maturity model for SaaS products (Figure 5). This model provides four levels of SaaS maturity.

In the first maturity level as depicted in the top left corner of figure 5 every tenant has its own unique product provided by and hosted at the service provider. This means configuration is provided by different versions of the code for each tenant. This level of maturity is similar to the Application Service Provider (ASP) model. SaaS can actually be seen as the successor of ASP (Buxmann et al., 2008; Kaplan, 2007). The ASP software model was introduced in order to serve the same purpose as SaaS does now; it also made use of the principle that the software provider hosts the software that the customer can use by connecting to it via the internet (Papazoglou, 2003). However, as explained above the ASP model was more expensive than the SaaS model and therefore quickly improved to evolve to the SaaS model.

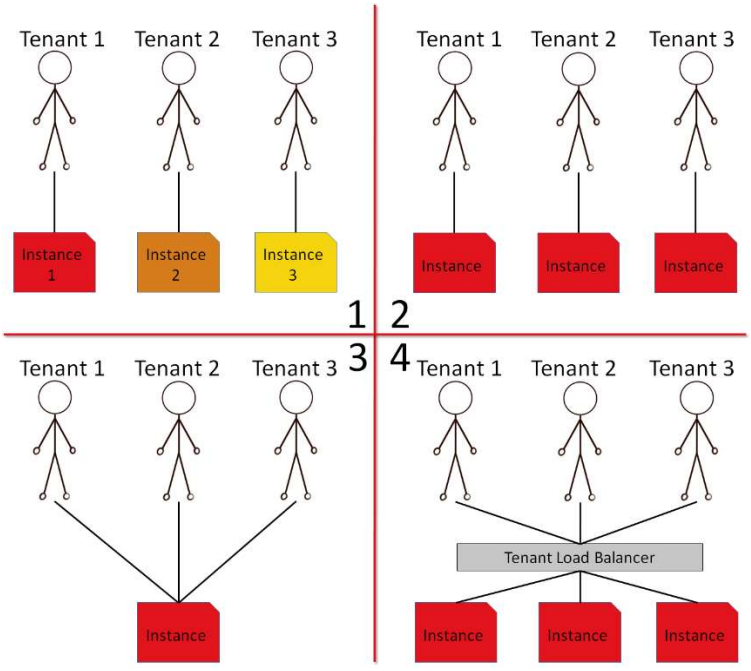

Figure 5. Software as a Service maturity model adapted from Chong & Carraro (2006).

In the next maturity level, depicted in the top right of figure 5, each tenant is provided with its own instance of the same code. Configuration is provided by offering configuration options so each tenant can customize its instance to their own preferences.

Where maturity level one and two can be seen as single-tenant architectures, maturity level three and four are using the so-called multi-tenant architecture. In the third maturity level, tenants make use of a single instance of the software product. Configurability is provided by making use of meta-data. The data of tenants is kept privately to each tenant. In order to make the multi-tenant architecture scalable a load-balanced farm of instances is hosted by the software provider, as can be seen in maturity level four.

In order to exploit the economy of scale the provider of SaaS should make sure their product adheres to the Multi-Tenancy architecture model (Mietzner, Metzger, Leyumann, Pohl, 2009; Bezemer et al., 2010b; Kwok et al., 2008; Kabbedijk et al., 2014). The multi-tenancy model is explained in more detail in section 3.3.

3.2 ERP as a Service

ERP Software fuses information throughout an entire enterprise to allow for a seamless flow of information (Engebrethson, 2012). This means ERP software provides companies with a way to manage their resources in an efficient and effective way (Fui-Hoon Nah, Lee-Shang Lau & Kuang, 2001). Resources can be material resources but also human resources and financial resources. ERP systems enable companies in doing so "by providing a total, integrated solution for the organization's information-processing needs." (Fui-Hoon nah et al., 2001). Since the year 2002 ERP vendors are recognizing and responding to the needs of companies to add BI functionality to their ERP systems (Griffin, 2007). This BI functionality is usually not as sophisticated as in standalone BI systems. (Chou, Tripuramalla & Chou, 2005) However, it suffices in providing smaller companies in their business intelligence needs. In section 3.4 BI is discussed in a more elaborate way, in this chapter more information is provided to motivate what is classified as "not as sophisticated" BI exactly.

Traditionally ERP software vendors have focused mainly on the large company segment of the market (Everdingen, Hillegersberg & Waarts, 2000). This focus has definitely borne fruit, since ERP systems are adopted by large enterprises up to a level of market saturation (Bernroider, Koch, 2001). Because of this market saturation ERP vendors are now focusing on SMEs (Haddara & Zach, 2011). SMEs feel the need for using ERP systems (Xu, 2011). However, as mentioned in previous sections of this thesis SMEs usually do not have a big IT budget and can therefore not afford the initial purchase of classic on-premises ERP software. However, if vendors offer an ERP that can be obtained by paying for example a monthly subscription fee, SMEs are able to obtain an ERP system even with their small IT budget. The way in which ERP vendors can do this is by offering a SaaS ERP solution (Makkar & Bist, 2012). To keep providing the software interesting for the vendor the vendor should be able to exploit the economies of scale. As discussed in section 3.1 this can be realized by employing the multi-tenancy architecture, multi-tenancy is elaborated upon in the next section of this chapter.

3.3 Multi-Tenancy

In chapter 1 of this thesis multi-tenancy was already introduced briefly. However, in order to gain a clear understanding of what multi-tenancy is exactly existing literature on the topic has been reviewed in order to find a good definition. The most recent definition found in scientific literature is by Kabbedijk et al. (2014) who define multi-tenancy as:

"Multi-tenancy is a property of a system where multiple customers, so-called tenants, transparently share the system's resources, such as services, applications, databases, or hardware, with the aim of lowering costs, while still being able to exclusively configure the system to the needs of the tenant."

Although this definition has been constructed by performing an extensive literature analysis we do not fully agree on the definition. Kabbedijk et al. (2014) argue that transparency should be included in the definition because it would otherwise be too generally applicable. According to the authors, there would be no difference between the multi-user model and the multi-tenant model if transparency is not included in the definition. In our opinion, the configurability part of the definition is already distinguishing the multi-tenant model from the multi user model. The difference between the multi-tenant model and the multi-user model is explained by Bezemer et al. (2010a) as follows: In the multi-user model the users are also using a shared application and database instance, however they have little configurability options. In the Multi-tenant model the tenants are using a shared application and database instance and have the possibility to configure the application, even to an extend in which the workflow of the application may differ per tenant. Another definition, albeit for a Multi-tenant application instead of just multi-tenancy, is provided by Bezemer et al. (2010a):

"A multi-tenant application lets customers (tenants) share the same hardware resources, by offering them one shared application and database instance, while allowing them to configure the application to fit their needs as if it runs on a dedicated environment."

This definition is almost identical to the one by Kabbedijk et al. but in our opinion, it's more accurate because it leaves the transparently part out of the definition. Bezemer et al. also provide a definition for what a tenant is exactly. They define a tenant as:

"The organizational entity which rents a multi-tenant SaaS solution. Typically, a tenant groups a number of users, which are the stakeholders in the organization."

Multi-tenancy's main benefit for the service provider is cost reduction, because the service provider only has to deploy one instance of the software and their hardware can be used more efficiently because not every tenant needs its own set of hardware (i.e. Servers). In addition, the utilization rate of the hardware is increased drastically when multiple tenants share the same IT infrastructure (Warfield, 2007; Kwok et al., 2008). This in turn provides a benefit for the customer, especially for SMEs since they do not need the computational power of a dedicated server (Bezemer, 2014).

3.4 Business Intelligence

As explained in the introductory chapter and in section 3.2 of this thesis some multi-tenant software systems offer their tenants with BI functionality. Before going into detail about the use of BI in ERP software systems, BI is defined. BI is a broad term that has a different definition to different people (Jones, 2009). The meaning of this term depends on the background of the one defining it (Sharda, Aronson & King, 2008). In this research, we embrace the definition by Wixom and Watson (2012) who define BI as: "a broad category of technologies, applications, and processes for gathering, storing, accessing, and analyzing data to help

its users to make better decisions". Companies in large proportions are embracing BI. The pressure, which drives companies into embracing BI, is a result of today's competitive business climate. The present business climate drives businesses into being more responsive to the changing business environment. Especially SMEs are vulnerable to the fast changing business conditions (Tutunea, Rus, 2012). In order to be more responsive to this changing business environment, businesses are in need of real-time decision support. This support can be provided by implementing BI systems.

These BI systems provide companies with the benefit of having access to accurate information on demand. This information can for instance be business performance. A company's business performance can, among others, be expressed in the form of for example sales figures, profit margins or geographical data. This information can be used as a source of information when making important decisions. More briefly, these systems provide decision support. A business support system can help in providing faster and more accurate reporting, improved decision-making, improved customer service and an increased revenue (Thompson, 2004).

BI is one of the functionalities that was not affordable for SMEs before the SaaS business model became ubiquitous. On-premises BI solutions were complex and costly. With the SaaS business model, affordable BI solutions became available. Apart from appearing as standalone products, BI functionality was also added to business software such as ERP systems. In ERP systems, BI is often put into practice in the form of dashboards. Dashboards are data visualization tools that provide visual displays of important information, preferably on a single screen so that the information can be processed and explored quickly (Palpanas, Chowdhary, Pinel & Mihaila 2007). A dashboard can for instance help to visualize a trend in the data that is hard to see from the data itself by showing the data in a graph. Dashboards play a big role in a today's business environments. According to Gile (2006) a researcher at Forrester Research, more than 40 percent of the 2,000 largest companies in the world use dashboards.

The most important measure for a software systems success is if the system meets the user's expectations (Cheng, Atlee, 2007). In order to discover what a systems purpose is requirements engineering is used (Nuseibeh, Easterbrook, 2000). Especially in dashboards, the information displayed on the dashboard should be tailored to its user's requirements (Frank, Heise & Kattenstroth, 2009). The data used to create the dashboard can be tailored, but also the way in which it is presented can be tailored. To illustrate this, you can think of a small web shop ran by one person on the one hand and a big oil magnate that operates on a worldwide scale on the other hand. The big multinational is probably interested in sales figures per sales region. For instance, sales in Europe, sales in the USA and sales in Asia. Whilst the owner of the web shop only needs the company wide sales figures, since he is probably not selling enough products to be interested in geographical data. This example is an obvious one, but it illustrates the difference in information needs based on the type of company.

3.5 Customization & Configuration

As seen in the previous sections, ERP systems, Dashboards, and multi-tenant software have to be tailored to its end-users. Tailoring is needed in order to meet the needs of a large number of different customers (Summers, 2001). There are two approaches to reaching a tailored SaaS product, *configuration* and *customization* (Guo, Sun, Jiang, Huang, Gao, Wang, 2008).

Customization involves changing the software products source code. "A customized feature is a feature that has been tailored to fit the nee ds of a specific user" (Jansen, Houben & Brinkkemper, 2010). SaaS providers usually do not provide customization because customization is usually much more costly than configuration (Guo, et al., 2008). This is because the SaaS vendor needs to maintain the separate code instance for every tenant instead of maintaining one single code base.

Configuration in contrast to customization does not involve change in the source code of the software product. Configurability has less ability than Customization. It provides users with configurable options using a single code base. This is why configuration is the preferred way of offering tailoring in SaaS products.

In literature, some sources mentioning *the flexibility usability tradeoff* (which is also called the Flexibility/Specialization Principle) are found. This principle explains that if software gains in flexibility it loses in usability (Lidwell, Holden, Butler, 2010; Steinebach, Guhathakurta, 2009). *Usability* is defined by ISO as: "The effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments"

Flexibility can be interpreted in several ways; in this research the assumption by Guo et al. (2008) is followed, that the term flexibility refers to the exact same thing as tailoring. This means that the tradeoff statement can be interpreted as: A higher level of configuration in a software product decreases its usability. This leads to the hypothesis that if a user is provided with more options to configure, there is more functionality to look at which decreases the simplicity of the product. Less simplicity in turn can lead to decreased usability.

3.6 Web Usage Mining

The ultimate goal of any user-adaptive system is to provide users with what they need without them asking for it explicitly. "Web usage mining refers to the automatic discovery and analysis of patterns in clickstreams, user transactions and other associated data collected or generated as a result of user interactions with web resources on one or more websites". (Mobasher, 2007) The goal of web usage mining is to capture, model, and analyze the behavioral patterns and profiles of users interacting with a website or web application. The discovered patterns are usually represented as collections of pages, objects, or resources that are frequently accessed or used by groups of users with common needs or interests (Mobasher, Cooley, Srivastava, 2000). These patterns can be used for multiple purposes and one of them is creating personalized experiences in the web application.

Web usage mining is a sub discipline of data mining. This means the standard datamining steps as proposed by Fayyad, Piatetsky, and Smyth (1996) in the Knowledge Discovery in Databases (KDD) process can be followed whilst performing web usage mining. The KDD process is depicted in figure 6 below.

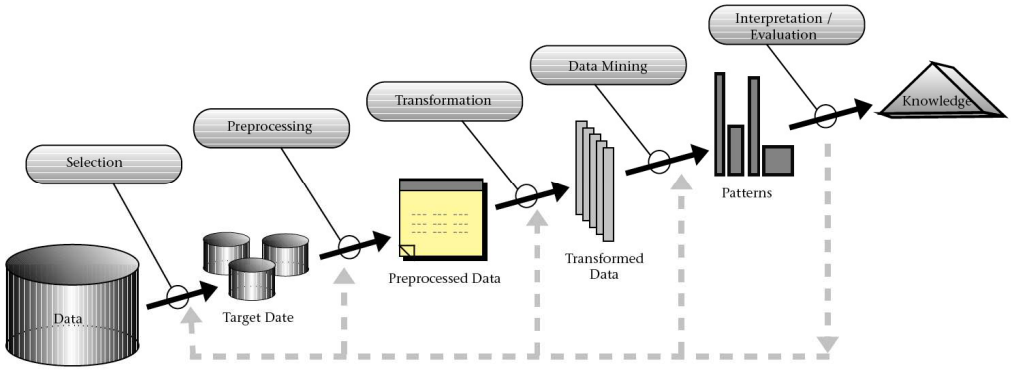

Figure 6. The KDD process as proposed by Fayyad, Piatesky, and Smyth (1996)

Pachidi, Spruit, and van de Weerd (2014), propose a detailed usage mining method as depicted in figure 7 below. This method adds a step before data selection called data understanding. This step is concerned with gaining an understanding of all the available data, and what this data can be used for. The next step is a combination of the first three steps of the KDD process, Pachidi et al. call the combination of data selection, data preprocessing and data transformation 'Data preparation'. The data mining step is split into two separate steps, namely exploratory analysis, and the in depth data mining step.

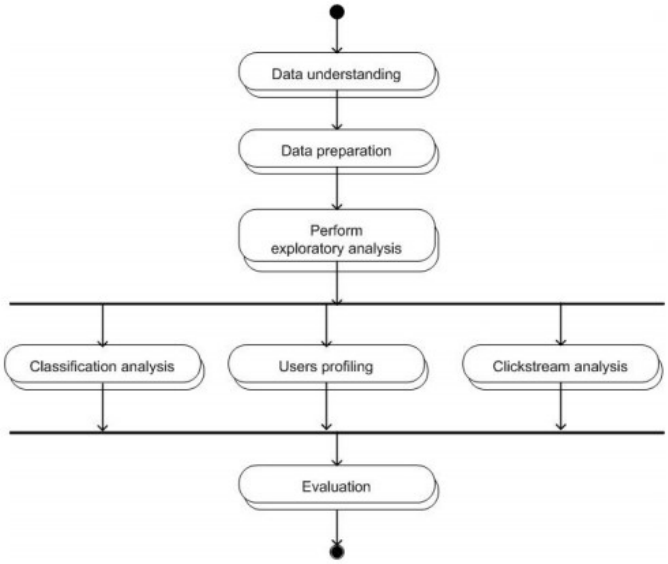

Figure 7. Usage mining method (Pachidi et al. (2014).

Liu (2007), Mobasher (2007), and Srivastava, Cooley, Deshpande, & Tan (2000), combine the first three steps in order to simplify the process to three main steps. These three main steps are: Data gathering and pre-processing, pattern discovery, and pattern analysis. These steps are depicted in figure 8 below. Altough the method by Pachidi et al. (2014) is more detailed it is similar to the steps as proposed by Srivastava (2000). In order to explain the web usage mining steps we follow the steps as proposed by Srivastava (2000) because the steps are more abstract and provide more room to explain the general concepts.

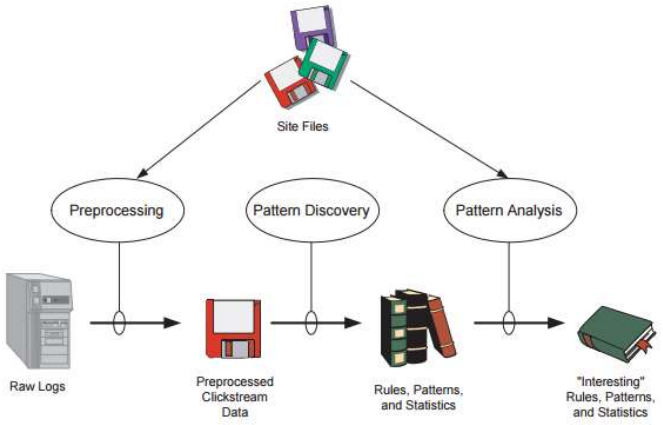

Figure 8. Web usage mining (Srivastava,2000)

3.6.1 Data gathering

The first step of web-usage mining is concerned with gathering the data that is going to be subject of the web usage mining and the pre-processing of this data. The data is typically gathered from multiple sources. The primary and most important type of data in web usage mining is usage data, without usage data there is no point in trying to mine web usage. However, usage data might not be enough for the data mining process. This can be solved by enhancing the usage data with other sources of data. The main categories of other data sources are: Content data, structure data and user data (Mobasher, 2007). The different types of data are explained briefly below.

Usage data is typically gathered from an activity log. An activity log stores the history of user transactions with a website/application. It usually contains information about what pages are being requested, the users requesting the pages, and a timestamp of when the pages are requested.

Content data is data about the content that is on the resources documented in the activity log. This information usually contains text or multimedia files such as images or videos.

Structure data is data that represents the application/website designer's view of the structure of the different resources in the usage data. This is usually represented in HTML and XML documents that contain the link structure of the application. Think of for example a menu structure XML file.

User data is extra data on users that is not stored in the activity log. This can be demographic data about the user, but it can also contain information about for instance the payment plan the user has at the moment.

3.6.2 Data Pre-processing

After gathering the necessary data from the available data sources, the raw-data has to be pre-processed. Usually this consists of data fusion, integration and cleaning, page view identification, user identification, sessionisation, and path completion. (Liu 2007)

Data fusion is needed when the usage data is stored on multiple locations. For instance, in big web applications the usage data is stored on multiple servers in order to reduce server load. Data fusion is the process of merging the data of the different sources of usage data together.

Data Integration is necessary if besides usage data other types of data are gathered. This can for instance be demographic user data, or semantic information on the webpages' contents or structure. These data sources need to be linked to the usage data so it can be analyzed together.

Data Cleaning is the process of removing irrelevant data from the data source. In the case of usage data this can for instance be page views that are not expressing any useful information for the analysis that will be performed.

Page view identification is the process of linking activity log entries to page views. This is important when an action in the activity log may represent multiple page views. This can be the case when a page is dynamic; the user can perform multiple actions on a page that are logged in the activity log as one action. This can also happen the other way around, in the case of websites that use multiple frames to show content multiple actions in the activity log may represent only one page view.

User identification is necessary when a website does not have an authentication mechanism. If a user does not have to login when using the website it is hard to identify the same user over multiple sessions. An approach that is used a lot for user identification is the use of client-side cookies.

Sessionization is the activity of identifying separate sessions of users on a website. This is done to find out what the sequence of page views of a user is during one visit to the website. If there is no authentication mechanism in place sessionization is usually performed by applying heuristics to the data. Heuristics that can be applied to usage data in order to recognize separate sessions can for instance be based on time. An example is that the time difference between two page requests may not be longer than 5 minutes. If the time between two consecutive page requests is longer than 5 minutes they belong to a different session.

Path completion can be necessary if the activity log is not complete. Depending on how the activity log is maintained certain page requests may not be stored in the activity log because of caching. This results in missing activity log entrances, in order to complete the user paths by adding these missing these missing entries heuristics can be applied. A heuristic that has knowledge on the structure of the website can infer if there is a missing page between two page requests.

3.6.3 Pattern discovery

The pattern discovery step is the step in which a data mining method or technique is performed on the data sources gathered and pre-processed in the previous step. Mobasher (2007) names three general ways of data mining for personalization. These are clustering, association discovery, and sequential modeling. Srivastava et al. (2000) in addition to these three techniques propose three more techniques. These techniques are Statistical analysis, classification and dependency modeling. Statistical analysis is concerned with descriptive analyses such as frequency, mean and median, we discard this as a pattern discovery technique because it is too general. Classification is only suitable for research in which predefined categories are available; this is not the case in this research. Dependency modelling on usage transactions is essentially association rule mining. Therefore, this section briefly explains the three techniques as mentioned by Mobasher (2007).

Association Rule Mining

Association rule mining is a data mining technique for discovering regularities between items in a dataset. A widely used example to illustrate association rule mining is that of shopping basket analysis in a web shop. An example of an association rule in this scenario is the following: “Users that buy product A and B also buy product C”. More formally, this rule can be expressed as A, B => C. An association rule consists of an antecedent and a consequent. In the previous example “A, B” is the antecedent and “C” is the consequent. In order to select interesting rules out of the full set of rules two constraints called support and confidence can be added (Mobasher et al. 2000). These measures express how valuable a rule is. Support gives the frequency of the association rule occurring in the data set. In the web shop basket example, this can be added to the rule by stating 50% of the transactions in the web shop consist of users buying product A and B. Confidence gives the probability that a rule holds. So in our example this can be added to the rule by stating for instance: “90% of the users that buy product A and B also buy product C”. This means that we can be 90% sure that if a user buys product A and B he will also buy product C.

Sequential modeling

Sequential pattern mining is a technique that, in the web-mining context, can find different paths users follow between pages of a website/web application. The sequential patterns represent the order in which pages are frequently viewed by multiple users.

Another term used for sequential modeling is clickstream mining (Pachidi et al, 2014). The main difference with association discovery is that sequential modeling takes the order of actions into account, where association discovery does not do this.

A method of sequential modelling is process mining. Process mining is a research field that can be positioned between machine learning & data mining on one side and process models on the other side. The goal of process mining is to use event data to extract process related information from an information system. (van der Aalst, 2011). Process mining focusses on all activities that use event data to extract process information. The goal is to discover or generate a process model by analyzing recorded events in an information system (van der Aalst, 2011). This can only be done when assuming that each event refers to a well-defined step in a process, and each event refers to a case (van Dongen, de Medeiros, Verbeek, & Weijters, 2005). In addition, events can have an actor that is responsible for executing the process step, and can have timestamps. Using these process models, it is possible to check conformance and recommend process redesigns. It is a way to analyze the complex processes and massive amounts of data that underlie these processes (van der Aalst, 2011).

For organizations, it is important to use event data in a meaningful way. Meaningful depends on the purpose, but some examples are the identification of bottlenecks, providing insights, or streamline processes (van der Aalst et al., 2012). Process mining activities can be categorized in several types, known as discovery, conformance, and enhancement (de Medeiros, Guzzo, Greco, van der Aalst, & Weijters, 2008; Song, Günther, & van der Aalst, 2009; van der Aalst, 2011). Discovery techniques take an event log, analyze it, and produce a process model without using any additional information (van der Aalst, 2011). Conformance entails the use of an existing process model, and compare this model with an event log of the same process. This technique can be used to check if the model is in line with the real process, and the other way around. The last type is known as enhancement, which focusses on the extension or improvement of an existing process model by using actual process information from an event log.

Clustering

Clustering is a broad term that can be applied in various ways. The ultimate goal of clustering is always to group a set of objects in a way that objects in the same cluster are more similar to each other than objects in another cluster.

An example of a clustering method is Formal Concept Analysis (FCA). FCA is a hierarchical clustering method. It is also described as a mathematical theory for creating concepts and concept hierarchies (Stumme, 2009). A concept hierarchy can be formed from a dataset by using FCA. In order to perform FCA a formal context is needed. A formal context consists of a set of objects and a set of attributes. Ganter, Stumme & Wille (2002) define a formal context as:

> *"A triple K:= (G, M, I), where G is a set whose elements are called objects, M is a set whose elements are called attributes, and I is a binary relation between G and M ($I \subseteq G \times M$). $(g, m) \in I$ is read 'object g has attribute m'."*

The formal context can be depicted in table form; the cells in the table contain ones and zeroes depending on whether an object has or does not have a certain attribute (Belohlavek, 2011). An example of a formal context is depicted in table 2. The goal of applying FCA is to create a formal ontology; this ontology can be created by applying FCA on the created formal context.

Table 2. Visual representation of an example formal context.

	Descriptor 1	Descriptor 2	Descriptor 3	Descriptor X
Object 1	1	0	1	0
Object 2	1	0	1	0
Object 3	0	1	1	0
Object N	0	0	0	0

Applying FCA to the formal context results in a set of formal concepts. For example, in table 2 object 1 and object 2 form a formal concept together because these are the only objects with the exact same descriptors. Ganter and Stumme (2002) define a formal concept as follows:

"For $A \subseteq G$, let

$$A' := \{m \in M \mid \forall g \in A: (g, m) \in I\}$$

and, for $B \subseteq M$, let

$$B' := \{g \in G \mid \forall m \in B: (g, m) \in I\}:$$

A (formal) concept of a formal context (G, M, I) is a pair (A, B) with $A \subseteq G$, $B \subseteq M$, $A' = B$ and $B' = A$. The sets A and B are called the extent and the intent of the formal concept (A, B), respectively. "

From the set of formal concepts, a concept lattice can be formed which is a visual representation of the relations between the formal concepts. The formal definition of a concept lattice by Ganter and Stumme (2002) is:

"The concept lattice of (G, M, I) is the set of all formal concepts of (G, M, I), ordered by the subconcept-superconcept order defined as:

$$(A1, B1) \leq (A2, B2) :\Leftrightarrow A1 \subseteq A2 \ (\Leftrightarrow B1 \supseteq B2)"$$

If the formal concepts for the example formal context in table 2 are calculated, it results in four formal concepts. These formal concepts can be visualized by creating a concept lattice. The concept lattice for the formal concepts calculated from the formal context in table 2 is shown in figure 9 below. Each circle in the lattice represents a formal concept. In this case, there are four different concepts. The formal concept on the top contains the objects that have the attribute "Descriptor 3" which in this case is not the case for any object, so this concept is empty. The bottom concept contains the objects that have all attributes, which again is not true for any object. The other two concepts are the concept {Descriptor 1, Descriptor 3} containing Object 1 and Object 2 and {Descriptor 2, Descriptor 3} containing Object 3.

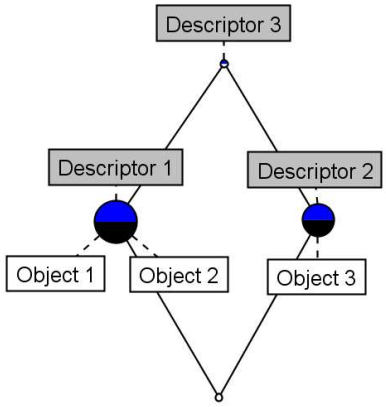

Figure 9. Concept lattice of the formal context in table 2.

3.6.4 Pattern Analysis

The final stage of web usage mining is pattern analysis. The goal of pattern analysis is to interpret the output of the pattern discovery stage. By interpreting the output of the pattern discovery stage useless patterns can be discarded. The resulting, useful, patterns can in turn be applied in order to satisfy the goals of the web usage-mining project. This can for instance be the restructuring of a websites' menu structure, or as in this research for personalization purposes. Depending on these project goals pattern analysis can be applied in a large number of ways.

4. Model Development - Data Gathering and Pre-processing

For the development of the model the web usage mining method, as described in section 3.6 of this document, has been applied. In the coming three chapters the process followed in this research is explained based on the four main web usage-mining steps described in section 3.6. These four main steps are: Data gathering, Data preprocessing, Pattern discovery, and Pattern analysis. A visual representation of these steps can be seen in figure 10. Data gathering and preprocessing are discussed in respectively section 4.1, and section 4.2. The methods considered for pattern discovery are discussed in chapter 5. In chapter 6 the technique used for pattern discovery is described in detail, this leads to a method that can be applied in other ERP systems in order to find different usage patterns of users. In chapter 7 Pattern analysis is discussed. The pattern discovery activities lead to an extended version of the PDD presented in chapter 6. The PDD in chapter 7 extends the previous PDD by adding steps that are specific for the generation of dashboards based on usage patterns. This PDD combines the web usage-mining steps that have been performed, from Data Gathering to Pattern Analysis.

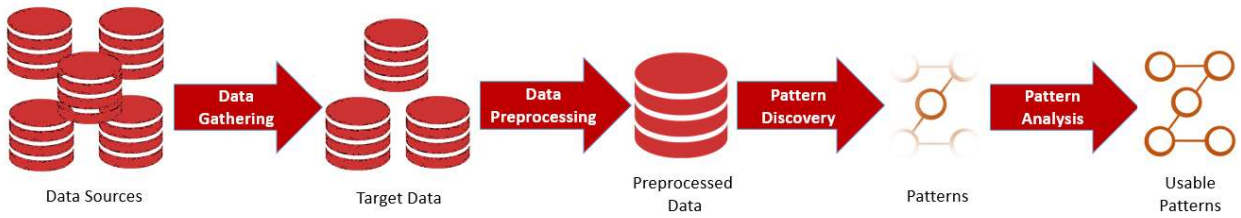

Figure 10. The web usage mining process

4.1 Data Gathering

For the development of the model, all available data sources available at the case study company have been gathered. The case study company provides access to the main types of data sources, namely: Usage data, Content data, Structure data and User data.

4.1.1 Usage Data

The usage data has been retrieved from the activity log of a big SaaS ERP system. The activity log contains usage data of the 250,000 customers of the ERP software product starting from November 2015. The database is still being expanded on a daily basis with around 12,000,000 new entries a day. An example of how a data entry in this database may look is given in figure 11 below. This shows that the author of this thesis on the third of March 2016 at 10:07 looked into the activity log (LgsActivitylog.aspx). This example is used in order not to reveal any real customer data. The data from the activity log is stored in a slightly different format in a SQL database. The transactions in this database consists of, among others, the following information: An accountID, a userID, an aspx and a timestamp.

Start date + time ▼	Application	User	Company
03-03-2016 10:07:18	LgsActivitylog.aspx	Jasper van de Wiel	Exact Netherlands B.V.

Figure 11. An example data entry in the activity log

4.1.2 User Data

Data on the users of the ERP system has been gathered by querying the contract database at the case study company. This database contains extra information on the customers of the SaaS ERP product. This database has been queried in order to get information on: The size of the company of the user in terms of the number of employees, the sector the company is operating in, the region the user lives in, and the package the company of the user is using. The full list of this data cannot be included in an appendix because of user privacy, a snippet of this information can be found in figure 13 below.

4.1.3 Content Data

The content data has been gathered from the ERP system by scraping the aspx'es of the ERP system for their title element. The scraper used for this process has been developed in python making use of the scrapy framework. Part of the resulting data is shown below in figure 12. The complete list of aspx'es and their title elements can be found in appendix A.

FinYearPeriods.aspx?	Overview: Financial years
FinYearPeriodsGen.aspx?	Create: Financial year
GenTopicChildCheck.aspx?	Template: Check
GenTopicLinkAdm.aspx?	Link existing template
HRMAbsenceGantt.aspx?	Leave registration per employee
HRMAbsencePercentages.aspx?	Absence percentages: Overview
HRMAbsenceRegs.aspx?	Overview: Leave registrations
HRMAbsenceTypesCLA.aspx?	Overview: Types of leave
HRMCostCenters.aspx?	Overview: Cost centres
HRMCostUnits.aspx?	Overview: Cost units
HRMDepartments.aspx?	Overview: Departments
HRMDivCategoryNames.aspx?	Overview: Company - classifications
HRMDivisionNew.aspx?	Company - New
HRMDivisions.aspx?	Overview: Companies
HRMEmplAnniversaries.aspx?	Overview: Anniversaries
HRMEmplBirthdayList.aspx?	Birthday list
HRMEmplContracts.aspx?	Overview: Employment contracts
HRMEmployees.aspx?	Overview: Employees
HRMIllnessProgRecover.aspx?	System Error

Figure 12. A snippet of the aspx'es and their title element

4.1.4 Structure Data

The structure data has also been retrieved by using a scraper. This scraper has also been developed in python, making use of the scrapy framework. For the structure data, the menu structure XML file of the ERP system has been scraped. From this file the menu item title, submenu item title, and the menu section title each aspx belongs to, are gathered. A snippet of the resulting data can be seen in figure 13 below. The complete list of aspx'es and the menu items they belong to can be found in appendix A.

Category	Subcategory	Heading	Aspx
Sales	Sales Orders	Sales invoices	LogEntries.aspx
Sales	Items	Items	InvTransactions.aspx
Sales	Sales Orders	Sales invoices	FinInvoices.aspx
Purchase	Returns	Returns	PurReceiptSelect.aspx
Purchase	Goods receipts	Goods receipts	PurReceipts.aspx
Purchase	Returns	Returns	PurReceiptReturns.aspx
Purchase	Purchase orders	Purchase orders	PurPurchOrdersPrint.aspx
Purchase	Goods receipts	Goods receipts	PurPurchOrderSelect.aspx
Purchase	Purchase orders	Purchase orders	PurPurchOrders.aspx
Purchase	Purchase invoices	Purchase invoices	PurPurchInvoices.aspx
Purchase	Items	Reports	PurPurchAnalysis.aspx
Purchase	Purchase orders	Purchase orders	PurEntryPurchOrder.aspx
Purchase	Purchase orders	Purchase orders	PurBalanceList.aspx
Purchase	Inventory	Overviews	LogItemCostPriceChanges.aspx
Purchase	Warehouse transf...	Warehouse tran...	InvWarehouseTransfers.aspx
Purchase	Warehouse transf...	Warehouse tran...	InvWarehouseTransferDeliveries.aspx
Purchase	Warehouse transf...	Warehouse tran...	InvWarehouseTransfer.aspx
Purchase	Inventory	Stock counts	InvStokCounts.aspx
Purchase	Inventory	Overviews	InvStockValue.aspx
Purchase	Inventory	Overviews	InvStockPositions.aspx
Purchase	Inventory	Stock counts	InvStockCountNew.aspx
Project	Invoices	Invoices	SlsInvoicePlanningAccounts.aspx
Project	Entry	Entry	ProWeekTransactions.aspx
Project	Reports	Reports	ProTimeCostTransactions.aspx
Project	Entry	Entry	ProTimeCostApproval.aspx

Figure 13. Snippet of the combination of aspx'es with structure data

4.2 Data Pre-Processing

After gathering the available data the next step is to pre-process this data. As explained in section 3.6.2, there are multiple pre-processing activities. The techniques performed in this research are data integration and data gathering. The other techniques were not applicable to the available data sources.

4.2.1 Data Integration

The first step performed in order to combine the different data sources is data integration. Data integration has been realized by first loading the data sources into a SQL database. This results in a separate table in the database for each of the data sources. The next step is combining the data sources in various ways and storing these combinations into new tables. The combinations that are made in the data integration process are the following:

The usage data has been combined with the user data. This results in usage data that is enhanced with information on region, package, company sector, and company size. A snippet of this information is shown in figure 14 below. The column 'LatestCommPackage' represents the current package of the user, the Environment column shows what country the user is based in. Moreover, the CompSizeCode column shows the size of the company

LogDateTime	Environment	AccountCode	UserID	LatestCommPackage	App	CompSizeCode	IsAccountant
2016-06-23 11:59:13.930	NL	[redacted]	[redacted]	EOLPremium	MenuPortal.aspx	G	Entrepreneur
2016-06-23 12:14:34.727	NL	[redacted]	[redacted]	EOLBasic	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 12:10:41.737	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 12:13:10.633	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 12:15:15.053	NL	[redacted]	[redacted]	EOLPSAAdvanced	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 12:16:23.383	NL	[redacted]	[redacted]	EOLPSAAdvanced	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 12:10:47.377	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 11:55:45.010	NL	[redacted]	[redacted]	EOLAccountancyPremium	ProTimeCostEntry.aspx	G	Entrepreneur
2016-06-23 11:57:07.493	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProTimeCostEntry.aspx	G	Entrepreneur
2016-06-23 11:55:39.697	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProTimeCostEntry.aspx	G	Entrepreneur
2016-06-23 11:59:19.383	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 12:01:18.307	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 12:09:54.940	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 12:01:23.573	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 11:55:47.870	NL	[redacted]	[redacted]	EOLBasic	ProTimeCostEntry.aspx	G	Entrepreneur
2016-06-23 12:12:48.600	NL	[redacted]	[redacted]	EOLPremium	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 12:12:44.850	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 12:17:20.710	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 11:59:05.307	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProTimeCostEntry.aspx	G	Entrepreneur
2016-06-23 12:00:11.040	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	HRMAbsenceRegs.aspx	G	Entrepreneur
2016-06-23 12:09:30.097	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	HRMAbsenceRegs.aspx	G	Entrepreneur
2016-06-20 09:18:30.057	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProProjectCard.aspx	NULL	Entrepreneur
2016-06-20 09:12:58.143	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	SysPopupMemo.aspx	NULL	Entrepreneur
2016-06-20 09:12:40.940	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	MenuPortal.aspx	NULL	Entrepreneur
2016-06-20 09:12:46.393	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-20 09:12:43.017	NL	[redacted]	[redacted]	EOLBasic	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-20 09:12:43.003	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-20 09:12:42.613	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	Exact.Online.Web.Areas.CRM.Controllers.AccountL...	NULL	Entrepreneur
2016-06-20 09:12:43.003	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-20 09:18:15.977	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-20 09:18:08.900	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-16 13:49:05.880	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	SysPopupMemo.aspx	NULL	Entrepreneur
2016-06-16 13:48:50.753	NL	[redacted]	[redacted]	EOLPSPremium	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-16 13:48:50.753	NL	[redacted]	[redacted]	EOLPSPremium	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-16 13:48:50.753	NL	[redacted]	[redacted]	EOLPSPremium	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-16 13:48:57.973	NL	[redacted]	[redacted]	EOLPSPremium	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-16 13:50:24.067	NL	[redacted]	[redacted]	EOLPSPremium	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-13 09:23:35.640	NL	[redacted]	[redacted]	EOLPSPremium	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-13 09:23:25.763	NL	[redacted]	[redacted]	EOLAccountancyBasic	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-13 09:23:21.123	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-08 17:46:49.173	NL	[redacted]	[redacted]	NULL	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-08 17:46:49.173	NL	[redacted]	[redacted]	EOLPREMIUM	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-08 17:46:48.220	NL	[redacted]	[redacted]	EOLBASIC	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-08 17:49:49.253	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-08 17:46:51.783	NL	[redacted]	[redacted]	EOLAccountancyBasic	ProWeekTransactions.aspx	NULL	Accountant

Figure 14. Snippet of the user data table

The next combination of data sources is the combination of usage data with structure data. This results in structure enhanced user transactions. This means that menu items, submenu items and menu headers are added to the user transactions in the usage database. This is called a page keyword table; a snippet of this information is shown in figure 15 below. Not all aspx'es are contained in the menu, this means that a number of usage transactions is lost in this integration because they cannot be enhanced with structure data.

User	App	LogDateTime	Category	Subcategory
[redacted]	FinBalanceSheet.aspx	2016-06-26 21:42:24.980	Financial	G/L Accounts
[redacted]	FinBalanceSheet.aspx	2016-06-26 21:42:21.507	Financial	G/L Accounts
[redacted]	FinBalanceSheet.aspx	2016-06-26 21:42:17.370	Financial	G/L Accounts
[redacted]	FinJournalSelect.aspx	2016-06-26 21:46:09.540	Financial	Entries
[redacted]	QuoEntryQuotation.aspx	2016-06-26 20:36:59.993	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 21:09:49.773	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 21:09:19.380	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 21:12:21.763	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 21:13:02.123	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 23:00:26.273	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 23:00:07.620	CRM	Quotations
[redacted]	SlsSalesOrderPicklist.aspx	2016-06-26 23:41:31.737	Sales	Sales Orders
[redacted]	CflImport.aspx	2016-06-26 21:58:09.333	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 23:04:15.350	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 23:33:50.000	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:02:26.767	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:55:59.943	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:58:09.477	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 22:25:11.743	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 23:33:40.373	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 20:52:32.013	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:39:24.367	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 23:04:19.053	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 23:04:18.957	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:39:35.210	Financial	Bank
[redacted]	SlsSalesOrderPicklist.aspx	2016-06-26 23:41:11.347	Sales	Sales Orders
[redacted]	CflImport.aspx	2016-06-26 21:55:59.800	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 20:40:32.703	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:02:26.393	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:02:19.033	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:55:56.380	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:58:04.177	Financial	Bank
[redacted]	FinJournalSelect.aspx	2016-06-26 22:15:55.197	Financial	Entries
[redacted]	FinJournalSelect.aspx	2016-06-26 22:53:30.567	Financial	Entries
[redacted]	FinJournalSelect.aspx	2016-06-26 22:59:24.133	Financial	Entries
[redacted]	FinJournalSelect.aspx	2016-06-26 20:31:37.703	Financial	Entries
[redacted]	FinJournalSelect.aspx	2016-06-26 22:53:36.977	Financial	Entries

Figure 15. Page keyword table

4.2.2 Data Cleaning

After data integration, the data has been cleaned in various ways. The irrelevant users, aspx'es, and menu items, have been removed. The motivation for classifying users, aspx'es, and menu items, as being irrelevant is described below.

The first data to be cleaned is the usage data, gathered from the activity log. It has to be cleaned of actions that are irrelevant. Actions can be irrelevant because of multiple facts. These facts are shown in table 3 below, accompanied with an example and a reason. For every irrelevancy, there are multiple examples, in the table only one example is stated for each reason. A full set of irrelevant aspx'es is provided in appendix B. The aspx'es that are classified as not manual are not included in the appendix, this would unnecessarily clutter the appendix.

Table 3. Examples of irrelevant page views

Irrelevancy	Example	Reason
Express no information	Menuportal.aspx	These aspx'es turn up in the activity log for every user, because the functionality is general system functionality.
Not manual	restapi/entity/crm/accounts	This is a non-manual action performed by an app.
Too general	FinTransactions.aspx	This aspx has multiple functions depending on a parameter that is not stored in the activity log.
Related to Country Specific Legislation	AssCDioxide.aspx	This page is related to the specific Carbon Dioxide tax laws as they are in Belgium.
Related to license/master data or hosting & reseller accounts		These pages are not relevant for regular exact online users
Dashboards, Quickbooks, Reports		Thesepages are removed from the analysis, because these pages are directly related to dashboards.

The next data to be cleaned is the user data, gathered from the contract database. This data consists of a number of items being: The size of the company determined by the number of employees, the sector the company is operating in, the region the user lives in, and the package the company is subscribed to. However, this data has a big number of missing entries. In fact, the data on the company size, and the data on the company sector are so incomplete that both columns have been discarded in the data.

The region information is used to delete users not situated in the Netherlands or Belgium. Users that are not in these countries cannot be visited in order to include them in the case study and are therefore not relevant users to base the model on. Another scoping decision is not to look at all packages offered by the ERP vendor, in this research only the packages for manufacturing, production and professional services companies are taken into account. This means users that have an accountancy subscription are discarded. The ERP vendor also offers educational licenses for high schools and universities, these licenses are discarded because students are not representative for employees of SMEs.

After cleaning the usage and user data, the semantic data also has to be cleaned. The content data consisting of page title elements has some missing data. This means some of the missing, or wrongly scraped page titles have been completed manually. The missing page titles are caused by pages that do not have their title between HTML <title></title> tags, but between for instance <h1></h1> tags or by pages that have no title at all. Pages that expect a parameter when being invoked are the main cause for the wrongly displayed page titles. Because of the way the scraper is programmed it does not add parameters to page requests, this causes the title of the page to be: "System Error".

The structure data has not undergone any data cleaning operations because this data was already in the correct format for analysis. However, some submenu items are not present in the combined dataset of usage and structure data. This is because for some submenu items all related aspx'es have been deleted from the usage data because of reasons stated in table 3 above.

5. Model Development - Methods Considered for Pattern Discovery

After gathering and pre-processing the data, the data-mining task can be performed in order to discover patterns in the data. In this research, the patterns represent different usage patterns expressing the different information needs of users. As explained in section 3.6, Mobasher (2007) names three main methods for data mining on web usage data. For the development of the model, all three of these methods have been considered.

Association rule mining has been discarded in early stages of the research. For sequential modelling, process mining is a technique that has been looked at in detail. Eventually clustering has been used in order to develop the model. To be more specific, content enhanced clustering has been applied, using Formal concept analysis (FCA) as the clustering technique.

In this chapter, it is shortly elaborated why association rule mining has been discarded as methods for developing the model. Then process mining, and how it has been attempted to use it in this research, is explained. Lastly clustering is discussed. First, the attempts to apply k-means clustering are discussed briefly, next formal concept analysis and our approach to FCA are explained in detail. This chapter provides an answer to the first sub research question:

SRQ1: How can the different usage patterns of SME employees, while using ERP systems, be identified?

5.1 Association Rule Mining

Association rule mining has been considered as a technique to be used to create a model of different usage patterns. The idea is applying association rule mining on the activity log in order to find sequences of pages regularly followed by different users. Performing association rule mining on the usage transactions results in a large number of association rules with an acceptable confidence value. However, the support for these rules is very low. Table 4. below shows the number of strict association rules for the analysis on 3 abstraction levels of the dataset, along with their average support rates. The low support rate shows that there is little use for these rules in the expression of different usage patterns. Based on this data, it has been decided to look into a more sophisticated technique for data mining that provides better ways of interpreting the results.

Table 4. Summary of the Association Rule Mining Findings

Dataset abstraction level	Number of strict rules	Average support of strict rules
Aspx'es	200844	0.12%
Submenu items	1337	0.32%
Menu items	5	0.09%

5.2 Process mining

Process mining can provide us with an overview of different paths users take in the ERP system. These different paths can possibly represent groups of users with the same behavior and therefore the same information needs. Because the data is gathered from an activity log, all data necessary for processes mining is available. There are timestamps, cases (users) and events (aspx requests). For this research, the discovery technique of process mining is the most appropriate one because there is no existing model to improve, or to compare the event log with.

The two tools that are most commonly used for process mining are ProM and Disco. Because ProM has slightly more functionality this was the first choice of process mining software. However, the software is slow and unstable when confronted with a large dataset as the one used in this research. This is why Disco has been used to perform process mining.

The first step is loading the dataset, consisting of user transactions combined with user data, into Disco. When visualizing this data Disco shows a so called 'spaghetti model'. This complex figure is too overwhelming to provide any value to our research, see figure 16 below. However, the user data gathered from the contract database provides us with means to filter on extra attributes in Disco. The attributes the user data provides are: Region, subscription type, company size, and sector.

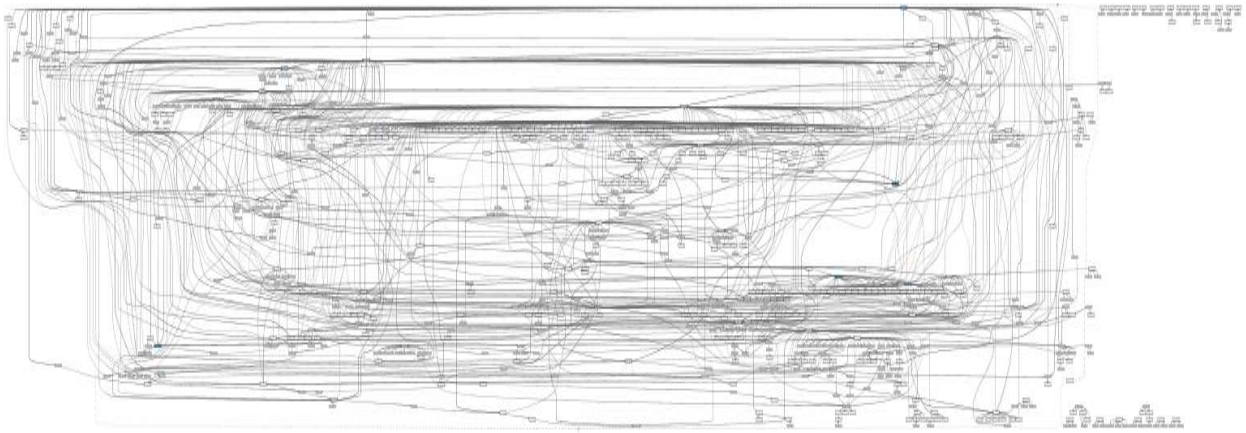

Figure 16. Disco Spaghetti model

Filtering enables looking at smaller parts of the dataset at once. This means the data can be filtered to only show the paths followed by for instance Dutch users with a manufacturing subscription. Applying these filters did simplify the models. One path used by a big group of users could be identified in these filtered models. For example, the path in figure 17 consists of only pages related to time entries. This means the users that follow this path only use the ERP system for time entries. However, other than this one specific path there are still too many paths between the different pages in order to identify different types of customers. There are some frequently followed paths, but these are used by all users, so they do not express any different information needs.This means a model of usage patterns cannot be created applying this method on the usage data combined with user data.

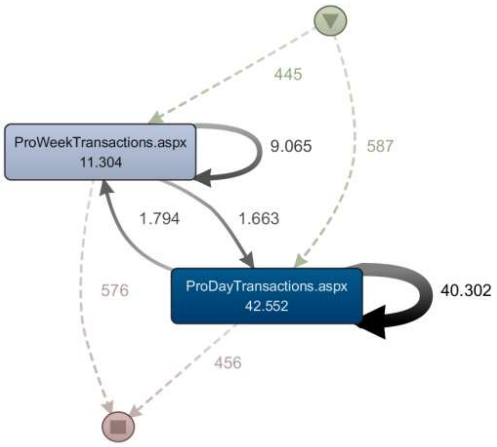

Figure 17. Path followed by time entry users visualized in Disco

However, another combination of data is still available. The usage data combined with the structure data is therefor also analyzed in Disco. Filtering on main menu item results in a clear overview, unfortunately this overview does not express any useful information. Using the submenu items to filter the data again results in a spaghetti model similar to the spaghetti model in figure 15 above. This leads us to the conclusion that process mining is not an appropriate technique to discover usage patterns from usage data.

5.3 Formal Concept Analysis

Formal Concept Analysis is a clustering technique that can help us by providing a concept lattice containing concepts consisting of users visiting the same pages within the ERP system. Users with the same usage patterns could possibly have the same information needs. The data needed to create such a concept lattice is available to us. The descriptors are the aspx'es in the ERP system and the objects are the user IDs from the activity log.

Before looking into formal concept analysis multiple clustering techniques have been explored. These techniques, including a basic k-means clustering approach were not powerful enough to find different usage patterns in our dataset. This section describes two different ways that have been used to apply FCA to the data. It is first explained how the preprocessed data has been transformed in order to be able to apply FCA on it. Then the first method of applying FCA is described, and after that, the second method of applying FCA is explained.

5.3.1 Formal Context Creation

The format of the usage data after data cleaning is not suitable for application of the FCA algorithm. The usage data is in the format of a big csv file with a single user action on every line. Moreover, as can be seen in table 2 above in the explanation of FCA, the formal context has to be of another format. The rows need to represent a user and the columns need to represent a certain aspx. The values in the table should be of a binary value representing if a user requested a certain aspx or not. Online, two conversion tools can be found that are developed in order to convert data into a format suitable for FCA. These tools are called FcaStone and FcaBedrock. Unfortunately, both tools froze when confronted with our dataset. This is most likely caused because the dataset contains too many rows for the software to handle. In order to get the data into the right format some data transformation operations have to be applied, including a pivot table.

Microsoft Office Excel is not a suitable solution in order to perform these data transformations. At least not on a dataset the size of the one in this research. The worksheet limit of an Excel file is 1,048,576 rows by 16,384 columns, and the data set in this research exceeds that number by far. This is why a python script has been created that performs a pivot table on the output file. The python script makes use of the pandas library, this is a library specifically created for the purpose of data analysis activities. With this script, the dataset has been transformed into the right format. Part of the formal context can be seen in figure 18.

UserID	AccGLJournalOverview.aspx	CflExpensesReceiptsEntry.aspx	CflImport.aspx	CflMatch.aspx	CflPayments.aspx
00F18	0	0	0	0	0
03F35	0	1	0	0	0
049F8C	0	0	0	1	0
08C06	1	0	0	0	0
0AF2C	0	0	0	0	0
0BC8A	0	0	0	0	1
0BCB4	0	0	0	0	0
0C944	0	0	1	0	0
0DD95	0	0	0	0	0
0F5BA	0	0	0	0	0
11178	0	0	0	0	0
12917	0	0	0	0	0
181AE	0	0	0	0	0
19B02	0	1	0	0	0
19D76	0	0	0	0	0
1B2B9	0	0	0	0	0
1BD21	1	0	0	0	0
1C0D6	0	0	0	1	0
1C4AE	0	0	0	0	0
1DD3A	0	0	0	0	0

Figure 18. Part of the formal context

5.3.2 Formal Concept Analysis

For the execution of the FCA algorithm, a number of tools has been explored and tried out. The first two options explored are java tools called ConExp and In-Close. Both tools are not able to handle the dataset and freeze when trying to import the data. The next option looked into is a python script called concepts.py. This script behaves more stable than the java tools when confronted with a small sample of the dataset. This script looked promising, also because the output format can be imported into graphviz, which is a nice tool to visualize the concept lattice. However, after running the script on the complete dataset for over 200 hours, it still did no output anything. In the documentation of the script the authors state “If you want to apply FCA to bigger data sets, you might want to consider other implementations based on more sophisticated algorithms like In-Close or FcbO” As stated above In-Close has already been tested, and did not work.

Logically FcbO was the next option. FcbO did indeed work with the dataset, and after 36 hours of running this script it delivered an enormous output. Unfortunately, this output file, in combination with the form of the output, is so complex it is impossible to make any useful observations from the data. This leads to the conclusion that performing FCA on only the usage data is not feasible. The reason for this is that the technique is computationally infeasible on such a big and diverse dataset.

Table 4. Different FCA tools and their performance on our dataset

	Import	Output	Usable Output
ConExp	No	No	No
In-close	No	No	No
Concepts.py	Yes	Yes	No
FcbO	Yes	Yes	No

5.3.3 Enhanced Formal Concept Analysis

However, the data gathered in the data gathering and pre-processing step can help enhance the usage data. The usage data can be combined with one, or multiple, of the other data sources. This idea is supported by a number of scientific papers that suggest adding semantic knowledge about page views to the data before performing any kind of data analysis, or to be more specific: Before clustering. This technique is called content-enhanced clustering.

Dai and Mobasher (2003) propose to use information associated with items on a website/application as a source of semantic knowledge. As explained in section 4.1 this data can be of two forms. On the one hand, there is content data containing for example text snippets from the pages. On the other hand, there is structure data that represents the application designer's view of the structure of the different resources in the usage data. It has been decided to use the usage data combined with the structure data in order to create different abstraction levels in the data.

In order to do this, first a formal context has been created containing user actions on a menu item level. Next, a formal context containing user actions on a submenu item level has been created. The tables that result from these data integration actions are called content enhanced transaction matrices. The combination of the usage data and the structure data into a content enhanced transaction matrix enables us to look at the data from different levels of detail. This means separate sub-sets of the dataset can be analyzed one by one. After some testing, we managed to create a model of usage patterns using formal concept analysis on subsets of the data making use of structure data to create these subsets. The development of the model is described in the next chapter.

UserID	Accounts	Bank	Budgets	Campaigns	Contact persons	Cost centres	Costs units	Currency	Entries	Entry	G/L Accounts	Goods receipts
FFFD57AC-...	1	1	0	0	1	0	0	0	1	0	0	1
FFFC78FE-...	1	1	0	0	1	0	0	0	1	0	1	0
FFFC76FC-...	1	0	0	0	1	0	0	0	1	0	0	1
FFEA872F-...	1	1	0	0	0	0	0	0	1	1	1	0
FFE51404-...	1	1	0	0	0	0	0	0	1	0	1	1
FFCF8041-...	1	0	0	0	0	0	0	0	0	0	1	1
FFCB9BCC-...	1	0	0	0	1	0	0	0	1	0	1	0
FF789586-...	1	1	0	0	0	0	0	0	1	1	1	0
FF736103-...	1	1	0	0	0	0	0	0	1	1	1	0
FF6FD805-...	1	1	0	0	1	1	0	0	1	0	1	1
FF58056B-...	1	1	0	0	0	0	0	0	1	0	0	0
FF44E05D-...	1	0	0	0	0	0	0	0	1	0	1	0
FF418879-...	1	1	0	0	1	0	0	0	1	0	1	0
FF11F397-...	1	1	0	0	0	0	0	0	1	0	0	1
FF0BDF45-...	1	1	0	0	0	0	0	0	1	0	1	0
FEF7382B-...	1	1	0	0	0	0	0	0	1	0	0	1
FEDD0723-...	1	0	0	0	0	0	0	0	1	0	1	1
FE8DAB06-...	1	1	0	0	0	0	0	0	1	0	1	0
FE70719B-...	1	0	0	0	1	0	0	0	1	0	0	0
FE6E9A79-...	1	1	0	0	0	0	0	0	1	0	1	0
FE43892E-...	1	0	0	0	0	0	1	1	1	0	1	0
FE2EF1C2-...	1	0	0	0	1	0	0	0	1	0	0	0
FE128441-...	1	1	0	0	0	0	0	0	1	0	0	0
FE05239D-...	1	0	0	0	0	0	0	0	1	1	1	0
FDF90606-...	1	1	0	0	1	0	0	0	1	0	1	0
FDD6425D-...	1	1	0	0	0	0	0	0	1	0	1	0
FDD09C74-...	1	0	0	0	0	0	0	0	0	0	1	1
FDC269F2-...	1	1	0	0	0	0	0	0	1	0	1	0
FD4965F1-...	1	0	0	1	1	0	0	0	1	0	1	1
FD3DD0E7-...	1	0	0	0	1	0	0	0	1	0	0	1
FD360665-...	1	1	0	0	1	1	1	0	1	0	1	0
FD35151F-...	1	1	0	0	1	0	0	0	1	0	1	1
FD2FFA0F-...	1	1	0	0	0	0	0	0	1	0	0	0
FD2C8741-...	1	1	0	0	1	0	0	0	1	0	1	0
FD2BF93C-...	1	0	0	0	0	0	0	0	1	0	0	0
FD2483A3-...	1	1	0	0	0	0	0	0	1	0	1	0

Figure 19. Part of the content-enhanced transaction matrix

5.4 Chapter Conclusion

After looking into different techniques of pattern discovery, it can be concluded that content enhanced formal context analysis is a technique that can be used to prove that there are different usage patterns. This means this technique can be used in order to find different usage patterns. These usage patterns can be discovered from a combination of the usage and structure data collected in the data collection stage. From the two data sources a content-enhanced transaction matrix can be created, on which FCA can be performed. This answers the first sub-research question of this research: "How can the different usage patterns of SME employees while using ERP systems be identified?" In the next chapter, content enhanced FCA is used in order to discover usage patterns.

Table 5. Summary of the different mining techniques

Technique	Ease of Use	Interpretability of patterns	Interpretability of patterns after enhancing the data
Association Rule Mining	***	*	*
Process Mining	***	*	**
Formal Context Analysis	**	*	***

6. Model Development – Pattern Discovery

After a long search for a technique that can discover different usage patterns in the data gathered from the ERP system, the final method used for pattern discovery is content enhanced clustering, using formal concept analysis as the clustering technique. In order to discover and visualize different usage patterns, FCA is performed on the content enhanced dataset that has been discussed in the previous chapter. This means FCA is performed on sub-sets of the dataset. These subsets are created based on different levels of abstraction. These abstractions levels are based on the structure data scraped from the ERP system. The context data scraped from the ERP system is used later on in the Pattern Analysis phase of web usage mining. The pattern discovery actions described in this chapter result in a model of different usage patterns. This model answers sub research question 2:

SRQ2: What are the different usage patterns of SME employees while using ERP systems?

The development of the model is the final step of the pattern discovery phase in the web usage mining approach. The next phase in the web usage mining approach is pattern analysis. The pattern analysis performed on the model answers sub research question 2 and is discussed in chapter 7.

6.1 The Dataset

As stated in the previous section a dataset has been created by combining multiple data sources. These data sources are: User transactions gathered from the ERP system's activity log, and context information scraped from the menu structure. The user transactions used in the analysis span one month (4 weeks containing 20 working days and 4 weekends). Table 6 shows that the data contains usage statistics of 21.637 unique users with a total number of actions of over 15.5 million. The data of a month has been used in order to get a complete picture of the actions users perform, without the dataset getting too big to perform an analysis on it. A month is suitable because some actions only happen at the beginning of a month and some actions only happen at the end of a month. Another reason why this data is representable is that the number of actions in the timespan of our dataset is almost equal to the average of the actions per month over a whole year. As can be seen in table 7 below, the average number of actions per month is 289 million. Note that these numbers are much higher than the number of actions in our dataset because they include non-manual actions.

Table 6. Information on the used dataset

Timespan	Unique Users	Total Actions
30-05-2016 to 26-06-2016	21.637	15.542.569

The context information scraped from the menu structure of the ERP system consists of a hierarchy of main-menu items that have submenu items that in their turn have aspx'es. Part of this structure is shown in figure 14 in chapter 4. The combination of the usage data and the context information is called a page-keyword table; part of this page-keyword table is shown in figure 14 in chapter 4. The hierarchy in this page-keyword table makes it possible for us to perform FCA on multiple abstraction levels. Which is explained below in section 6.2.

Table 7. Overview of total number of actions per month

Year – Month	Total number of actions (Millions)
2015-11	241
2015-12	242
2016-01	322
2016-02	277
2016-03	287
2016-04	326
2016-05	265
2016-06	284
2016-07	309
2016-08	254
2016-09	296
2016-10	367
Average	**289**

6.2 Performing Content Enhanced FCA

The highest level of abstraction, that provides the least detailed view, is grouping the user's page views by the menu category each page belongs to. Performing FCA on this abstraction of the dataset provides us with the concept lattice as shown in figure 20. This concept lattice is created by performing FCA on a formal context created by using user IDs as objects and main-menu items as attributes. The main menu items are 'Purchase', 'Sales', 'Person', 'Project', 'CRM', 'Financial', and 'Manufacturing'.

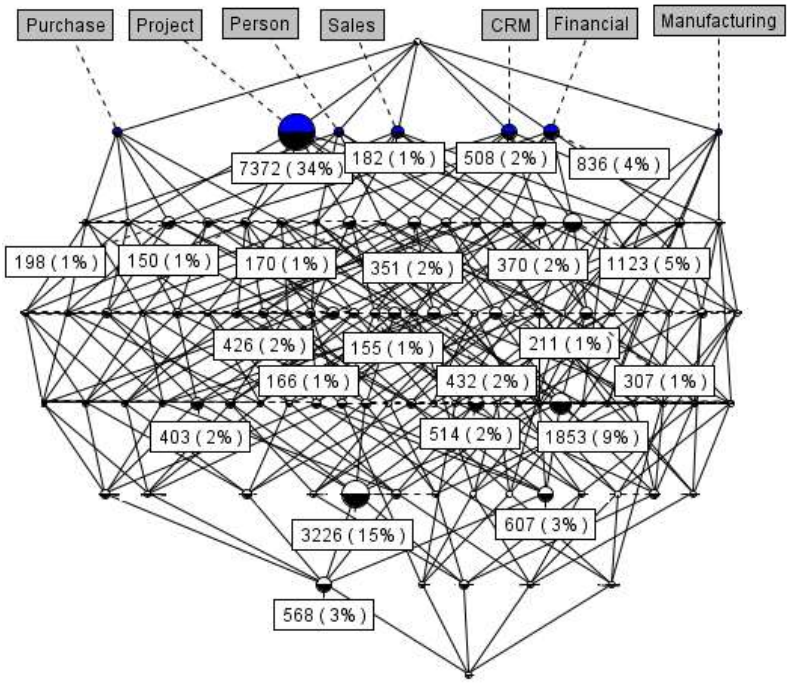

Figure 20. Concept lattice of page views grouped on main-menu item.

This concept lattice shows groups of users that visit pages with the same main menu item related to them. The notion of a concept lattice has been explained in chapter 3. In this lattice, the objects are users, represented by user IDs, and the attributes are main menu items. A new concept in this lattice are the labels that do not state the objects, but state a number and a percentage. This label is explained by looking at the biggest node in this lattice. This is the second node from the left, on top of the diagram. As can be seen this node groups the users that only view pages under the 'Project' menu item. This node is accompanied by a label stating "7372 (34%)". '7372' stands for the number of users that are in this node, so 7372 users only view pages that are grouped under the 'Project' main menu item. The percentage represents the percentage of users that are in this concept related to the total number of users in the dataset.

Not all the concepts in figure 20 are accompanied by a label. This is because the labels of concepts with less than one percent of the users in them have been removed, in order to keep the image clear. This means the 22 biggest concepts of this lattice are shown and discussed in this chapter (The concepts accompanied with a label in figure 20). This accounts for 95% of the users of the ERP system and their actions in the month July (30-05-2016 to 26-06-2016). The other concepts also express different usage patterns, but for the explanation of our pattern discovery technique and the model development, the 21 biggest concepts give a clearer overview.

For clarification table 8 below numbers the 22 concepts. It also provides information on the attributes in each concept, the number of users in the concept, and the corresponding percentage in terms of the total number of users. The newly introduced concepts and sub-concepts are clarified in similar tables later on. Each (sub-) concept has a unique number, which is used to refer to the concept throughout the complete document. The percentage in the last column is always expressing the percentage of users in relation to the total number of users in the (sub-)concepts' super-concept. Not every concept is shown in this document. All concepts, lattices and explanatory tables of the model can be found in the appendices of this document.

Table 8. The 22 biggest concepts on Category level

Concept #	Attributes	Users	Percentage
1	Project	7372	34%
2	CRM, Financial, Project ,Purchase, Sales	3226	15%
3	CRM, Financial, Project, Sales	1853	9%
4	CRM, Financial	1123	5%
5	Financial	836	4%
6	CRM, Financial, Person, Project, Sales	607	3%
7	CRM, Financial, Person, Project, Purchase, Sales	568	3%
8	CRM, Financial, Purchase, Sales	514	2%
9	CRM	508	2%
10	CRM, Purchase, Sales	432	2%
11	CRM, Project, Sales	426	2%
12	CRM, Sales	403	2%
13	CRM, Project, Purchase, Sales	370	2%
14	Person, Project	351	2%
15	CRM, Financial, Sales	307	1%
16	CRM, Financial, Project	211	1%
17	Purchase,Sales	198	1%
18	Sales	182	1%
19	Project, Sales	170	1%
20	CRM, Financial, Purchase	166	1%
21	Financial, Project, Sales	155	1%
22	Person, Project	150	1%

The fact that clusters of users that have similar usage patterns are expressed by the output of the formal concept analysis, means that FCA is a suitable approach to discover different usage patterns. However, the distinction of different usage patterns based on main menu items is not specific enough to link information needs to them in the pattern analysis step. Therefore, the next step is a drill down on these 22 concepts in order to get more detailed sub-concepts out of them.

The lowest level of abstraction, and therefore the highest level of detail, that can be reached are the single aspx'es. The abstraction level between the main menu categories and the aspx'es is looking at the users page views on the submenu item level. For the development of the model, the submenu item abstraction level has been considered. This level of abstraction has been chosen because the category abstraction level is too abstract, and therefore does not express enough information about different information needs of usage patterns. The aspx level on the other hand is too detailed to base usage patterns on.

An example of this drilling down is shown below in figure 21. Here the actions of the users in concept 4 as shown in table 8 have been enhanced with submenu item data instead of main-menu item data. The CRM menu item has three submenu items. These are 'Accounts', 'Contact Persons', and 'Quotations'. In addition, the Financial menu item has 4 submenu items. These are ''Bank', 'Currency', 'Entries', and 'G/L Accounts'. In reality, each main menu item has more submenu items but these have been deleted in the data-cleaning step as explained in chapter 5. Again, the labels of concepts with less than 1% of the total number of users in this lattice have been deleted from the image to get a clearer image. Again, a table of the biggest concepts in this lattice has been created. This table is shown below in table 9.

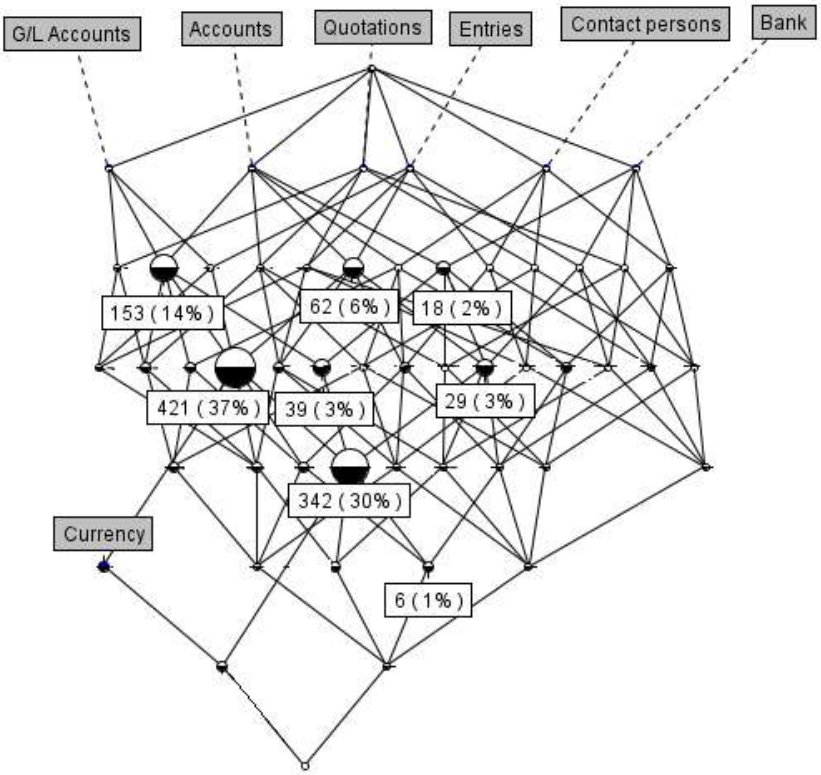

Figure 21. Concept lattice of concept 4.

Table 9. Sub-concepts of concept 4

Concept #	Attributes	Users	Percentage
4.1	Accounts, Entries, G/L Accounts	421	37%
4.2	Accounts, Bank, Entries, G/L Accounts	342	30%
4.3	Accounts, G/L Accounts	153	14%
4.4	Accounts, Entries	62	6%
4.5	Accounts, Bank, G/L Accounts	39	3%
4.6	Accounts, Bank, Entries	29	3%
4.7	Accounts, Bank	18	2%
4.8	Accounts, Bank, Contact Persons, Entries, G/L Accounts	6	1%

Drilling down one level deeper to aspx level gives us the concept lattice in figure 22. This lattice is already becoming hard, or even impossible, to read for the human eye. This level of abstraction also starts getting computationally hard, and does not express useful information. Because the submenu items already express a diverse set of usage patterns, it has been decided to base the model on the submenu item abstraction level. A concept lattice of the concepts listed in table 8 has been created on submenu item abstraction level. The concepts in these lattices have been combined together in a large table file in order to create a textual model. This led to ~1600 different usage patterns, these usage patterns are numbered in the same way the concepts have been numbered in table 8. In table 10 below, the example usage patterns 4.1, 4.2 and 4.3 are shown.

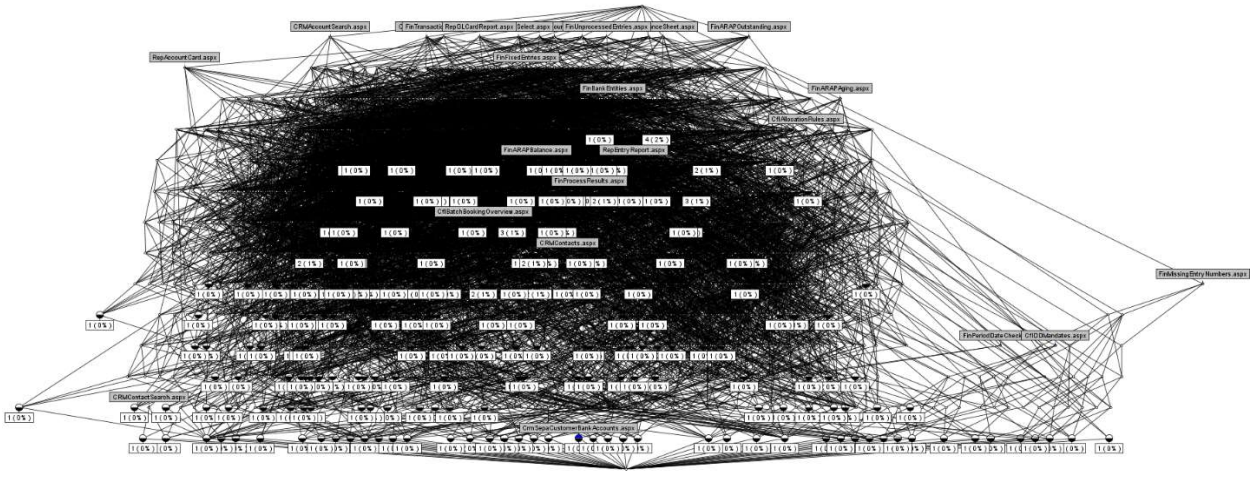

Figure 22. Concept 4, on page view abstraction level

Naming the patterns mentioned above makes no sense because there is no ground to base the naming on. A large portion of the usage pattern only differ from each other in only one or two submenu items. In theory there can be 2^19=524,288 different usage patterns, because there are 19 submenu items that are taken into account in pattern discovery. In addition, each submenu item can have two values, namely '1' (visited) or '0' (not visited). Therefore, it is decided not to use the textual model, as the final model created in the pattern discovery step. Instead, the model is expressed by visualizing the method followed to group the user behavior in to usage patterns in a step-by-step PDD. The PDD is accompanied by an explanation of each of the steps and its deliverable.

Table 10. Example of three usage patterns.

Subcategories	User Type 4.1	User Type 4.2	User Type 4.3	
Accounts		1	1	1
Contact Persons		0	0	0
Quotations		0	0	0
Bank		0	1	0
Currency		0	0	0
Entries		1	1	0
G/L Accounts		1	1	1
Employees		0	0	0
Payroll		0	0	0
Entry		0	0	0
Invoices		0	0	0
Projects		0	0	0
Planning		0	0	0
Inventory		0	0	0
Purchase Orders		0	0	0
Returns		0	0	0
Items		0	0	0
Sales Orders		0	0	0
Subscriptions		0	0	0

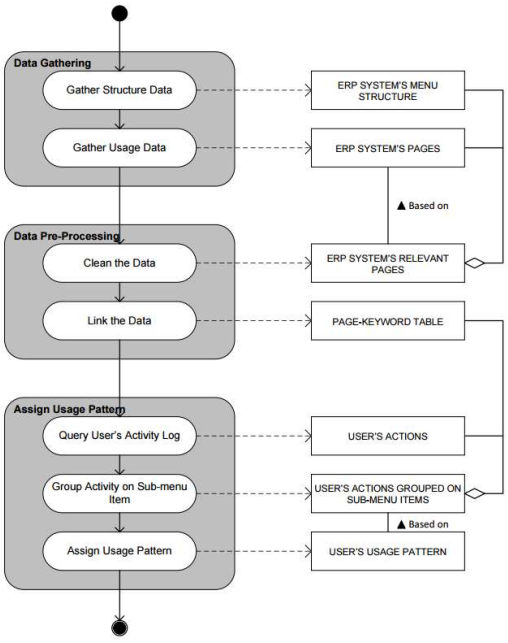

Figure 23. PDD of the usage pattern discovery method

This PDD shows how the model can be used to link a usage pattern to a specific user. The first activity is data gathering. This activity consists of two steps in which the structure data and the usage data are gathered from the ERP system. The structure data is gathered by scraping the menu system of the ERP system. The usage data is gathered from the activity log of the ERP system. The deliverables linked to these two steps are the ERP system's menu structure, and the ERP systems pages. The usage data contains more information, but for the creation of the page-keyword table, only the pages that the ERP system consists of are needed.

The next activity is the data pre-processing, this activity consists of two steps. The first step is data cleaning; in this step the relevant pages of the ERP system are selected from the usage data. What pages are relevant is dependent on what the usage patterns are going to be used for. For example, in this research the login page is discarded because it does not express any value for the creation of personalized dashboards. The next step is linking the cleaned usage data to the ERP system's menu structure so that each relevant page is linked with a menu item and if applicable a submenu item. This results in the Page-Keyword Table. The Page-keyword table can be loaded into a relational database.

The third and last activity is assigning usage patterns to specific users. This activity consists of three steps. First the user's activities in the past 30 days are queried. These are stored in a different table in the relational database that the page-keyword table is also stored in. Next the users' actions table is joined with the page-keyword table. With a count statement on the number of visits to a submenu item. This leads to a view that shows the users' actions grouped on how many times the user visited pages under a certain submenu item. The last step is deducting a usage pattern from this table. In this research, a threshold of five visits per submenu item has been used, to count a submenu item in the usage pattern. The usage pattern consists of a string of 1's and 0's based on what subcategories a user has visited. A one means that pages belonging to this submenu item have been visited more than the threshold value. A zero means pages belonging to this submenu item have been visited less than the threshold value.

The usage pattern string, which is determined by applying the steps of the model, can now be used in order to create personalized information in the system. In this research, the usage pattern is specifically used to generate personalized dashboards. Therefore, information needs are linked to the submenu items. These steps are discussed in chapter 7. In this chapter some more steps, specific for the generation of personalized dashboards are added to the PDD.

6.3 Chapter Conclusion

From this chapter it can be concluded that content enhanced FCA can indeed be used to discover different usage patterns. These usage patterns can be based on the submenu item abstraction level. The concept lattices created using FCA give insights in how the different kinds of usage are distributed in percentages of the total number of users. It can be concluded that there are a big number of different usage patterns that can be generalized in order to group similar users together. However, the high number of different usage patterns, in combination with the slight differences between those users' types makes it impossible to give the usage patterns meaningful names. Also because of the aforementioned high number of different usage patterns, a textual model of these different usage patterns becomes hard to grasp because of its size. It is therefore decided to express the model of different usage patterns in the form of a method. Using this method enables linking a usage pattern to every user of the ERP system by looking at their activity data.

The method expressed by the PDD in figure 23, provides a way to answer our second research question: *"What are the different usage patterns of SME employees while using ERP systems?".* In this document, the question is only answered for the ERP system developed by the case study company of this research. However, with the PDD provided in this chapter the method can be repeated at other ERP software companies as well, in order to discover the different usage patterns of SME employees using that specific ERP system.

7. Model Development – Pattern Analysis

After pattern discovery, as discussed in chapter 6, the next phase in the web usage approach is Pattern Analysis. In this research, the goal of pattern analysis is analyzing how information needs can be linked to the different usage patterns, and linking dashboard components to these information needs. The result of the pattern analysis stage answers the third sub research question:

> ***SRQ3**: How do the usage patterns express different information needs of SME employees while using ERP systems?*

7.1 Pattern Analysis

The first step of pattern analysis, linking information needs to the usage patterns, is realized by using the content information. This information has been scraped in the data gathering & preprocessing phase. This data consists of the titles of the different aspx'es in the ERP system. These titles give a global description of what information is on the different pages. This content information gives us a global idea of what information a user is looking for on the different pages, and thus what information needs the user has when using the ERP system. Part of the aspx'es and their titles is shown in figure 11 in the content data section of chapter 4.

In order to get an understanding of how the users' information needs can be converted into dashboard components the BI dashboards in multiple ERP systems have been studied. The ERP systems that have been studied for dashboard components are listed in table 11 below. This table also shows what subscription type(s) the dashboards in the system provide information for. The systems taken into account are either free, or could be informed for free by for instance using a free trial period.

Table 11. ERP systems that provided inspiration for dashboard components

Name	Manufacturing	Wholesale	Professional services
Beegua (dashboarding app)			X
Exact Online	X	X	X
Odoo	X	X	X
Qlikview for Exact Globe	X	X	X
Speedbooks (dashboarding-app)	X	X	X
Visionplanner (dashboarding app)	X	X	X

After linking dashboard components to the submenu items based on these two sources of information a number of experts within the case study company have been interviewed. These interviews have been conducted in order to evaluate if the dashboard components linked to the subcategories are relevant to the ERP system of the case study company. Based on these interviews, it has been decided to not include components on assets, campaigns, cost centres, cost units and work centres. After these interviews, the list of dashboard components has been finalized. The final list of components per submenu item is in table 12 below.

Table 12. The final list of possible dashboard components

Submenu item	Component
Accounts	Outstanding items Receivable (Aging)
Bank	Current balance on bank accounts
Contact Persons	List of contact persons (sorted on most recently contacted)
Currency	Exchange rates
Employees	Man/Woman ratio
	Birthdays
Entries	# of unprocessed entries
Entry	Hours per project
	Hours per week
	Billability
G/L Accounts	Revenue, Costs, and Result by Year
Inventory	Total inventory value
Invoices	# of invoice proposals
Items	Top items by revenue
Payroll	Contract types
	FTE Distribution
	Distribution Employees per division
Planning	(See Sales orders)
Projects	# of projects per manager
	# of projects this year/last year
	Project revenue/ this year /last year
Purchase orders	Number of purchase invoices current year
	Total amount in euros purchase invoices current year
	Distribution per GL account
	Distribution per supplier
Quotations	(See Sales orders)
Returns	# of returns
Sales orders	# of Open sales orders
	Amount of open sales orders
Subscriptions	Active subscriptions

As stated above this selection of components per submenu item has been made based on the title elements of the aspx'es belonging to these subcategories. In combination with the knowledge about existing BI dashboards. An example of this is explained below.

Under the submenu item ‘Entry’, fall three aspx’es; "ProDayTransactions.aspx", "ProWeekTransactions.aspx" and “ProTimeCostApproval.aspx”. From the table with aspx’es and title elements, the following three entries are retrieved:

Table 13. Aspx’es and their title, in the submenu item ‘Entry’

Aspx	Title element
ProWeekTransactions.aspx	Weekly entry
ProDayTransactions.aspx	Daily entry
ProTimeCostApproval.aspx	Approval

From this information alone, it can be concluded that the user is interested in daily & weekly entries, and approvals. Combining this information with domain knowledge enriches this conclusion with the fact that these entries have to do with time entries and approvals. Looking into other BI dashboards that have components that are related to time entries it is concluded that a component can be created showing the user entered hours per day or week. Another component often shown together with entered hours is billability. The billability also comes forward in the evaluation with domain experts. Based on the combination of these information sources, the three components shown in table 12 above are shown when a usage pattern contains the submenu item ‘entries’.

7.2 Expanding the model

With results of the pattern analysis stage of web usage mining performed in this chapter, some steps are added to the PDD of our model as presented in figure 23. These steps are specific for the generation of dashboards, based on the usage patterns discovered with the model as described in chapter 6. These extra steps are shown in the PDD in figure 24 below, and are be explained in the text in this section.

To the data gathering activity a step is added that is concerned with gathering content data, in this case this results in the ERP system’s page titles. The data pre-processing activity still has the same steps, but the data cleaning step now also concerns cleaning the content data

A completely new activity called ‘Link information needs’ is added to the PDD. This activity consists of four steps related to linking information needs. First potential information needs are linked to the different submenu items of the ERP system. In order to do this the ERP system’s page titles scraped in the data gathering activity are linked to the pages, by doing this an educated guess on what information is on the pages can be made. The next step is evaluating the potential information needs linked to the different pages of the ERP system with domain experts working at the ERP vendor. After that, potential dashboard components can be linked to the evaluated information needs. These components need to visualize the information that users need in order to fulfill their information need. This step also needs to be evaluated with domain experts working at the ERP vendor. This activity eventually leads to one or more dashboard components linked to each different submenu item

The ‘Assign Usage Pattern’ activity is still the same as in the PDD shown in figure 23. The only difference is that now the resulting usage pattern expresses dashboard components that can fulfill the user’s information needs. Based on this usage pattern and the related dashboard components a user specific dashboard can be generated in the ‘generate dashboard’ activity.

This method has been evaluated with 15 possible end users, using a prototype of the dashboard generator. The prototype is introduced in the next chapter. The evaluation is explained in chapter 9, and the results of the evaluation are discussed in chapter 10.

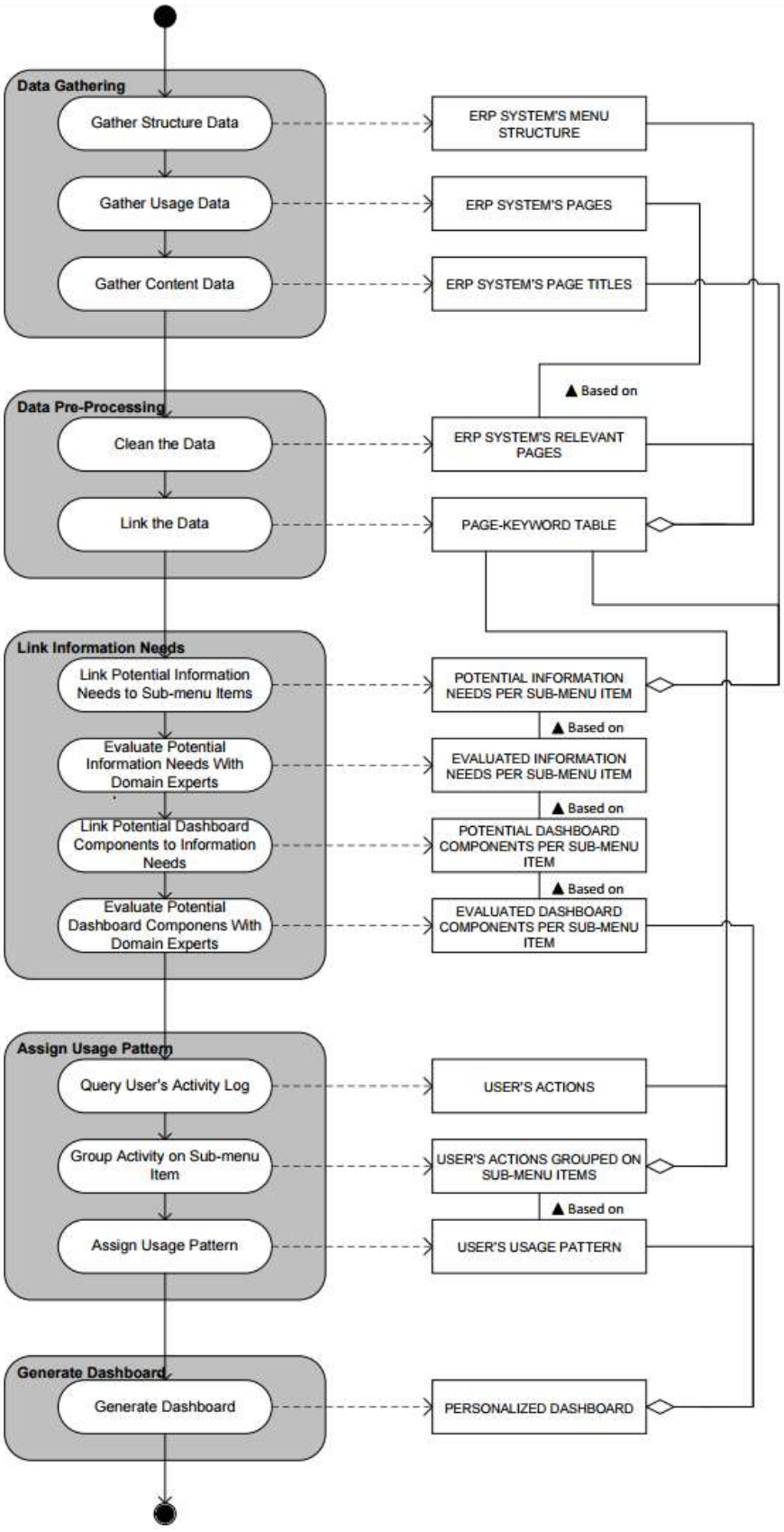

Figure 24. PDD of the dashboard generation method

7.3 Chapter Conclusion

The information needs of different users are hypothetically expressed by the information that is on the pages they visit. In this research the information on the different pages of the ERP system has been gathered by scraping the title elements of these pages. The information on these pages has been linked to dashboard components inspired by existing ERP dashboards, and these components have been evaluated with experts in the field. Linking these dashboard components to the usage data of the ERP system answers the third sub research question: ”How do the usage patterns express different information needs of SME employees while using ERP systems?” Each usage pattern expresses different information needs of users of the ERP system based on their ERP system usage. The last step in the research is now to evaluate the model, and the components linked to the different usage patterns in the model. This last part is completed by employing the prototype of a dashboard generator, as explained in chapter 8, in the case study that is described in chapter 9.

8. Prototype

The last step before the model can be evaluated is creating the prototype of a dashboard generator. This prototype employs the model in order to generate personalized dashboards to users of the ERP system developed by the case study company. The prototype has been developed making use of PHP and Microsoft Power BI. The decision to visualize the dashboards with Microsoft Power BI is made because this tool enables for the fast prototyping of dashboards. The prototype is designed to be a product that when refined is release ready for the case study company.

The first steps in the prototype are concerned with authorization, and the extract, transform, and load (ETL) process. Our case study company provides functionality that enables developers to integrate their application with the SaaS ERP Product. This functionality comes in the form of a public API. Each API request is authenticated using OAuth2.0. Users of the SaaS ERP product can give an app permission to access their ERP data. Figure 24 below show the authentication screen a user is prompted with after logging in to the dashboard application.

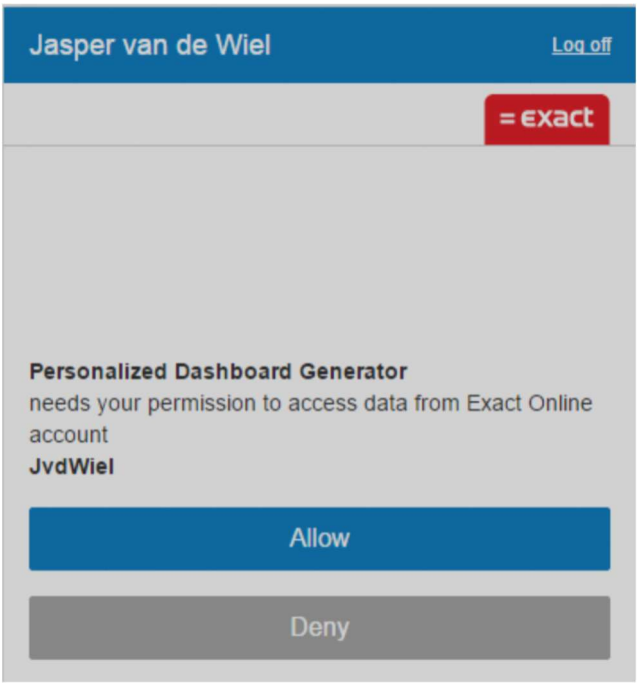

Figure 25. The apps authentication screen

Once the user logs in to the application and allows the application permission to access the users' data, the application has access to the users' User ID. The user ID is checked against a database containing the usage patterns of all EOL users based on their EOL usage of the last 30 days. As explained in chapter 6 the usage pattern is expressed in the form of a string of 1's and 0's based on what subcategories the pages the user has viewed belong to. The usage pattern of the user determines what dashboard components are shown on the dashboard and thus what data is extracted from EOL. As explained in chapter 7 the subcategories have been linked to one or more dashboard components. Each '1' or '0' in the usage pattern stands for a submenu item visited by the user, and a submenu item represents one or more dashboard components. A '1' means this component will be in the dashboard a '0' means it will not be in the dashboard. The data extraction is completed by performing API calls on the ERP system account of the user. The extracted data is stored in a SQL database.

The user is then prompted with a dashboard containing the components based on their usage pattern. Below four screenshots of dashboards are shown. In table 14, their underlying usage pattern as defined by the model explained in chapter 6, are shown. These dashboards are dashboards generated for participants of our case study (Explained in the next chapter). For reasons of confidentiality, the user data is replaced with demo data. The dashboards are in Dutch, because the case study participants have Dutch as their native language.

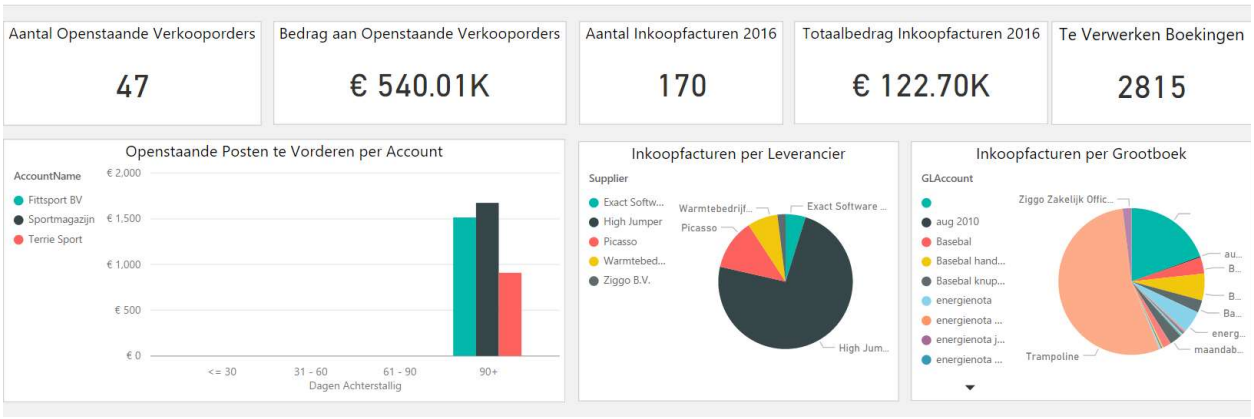

Figure 26. Dashboard for the first usage pattern shown in table 14

Table 12 shows how the usage patterns of four case study participants has been determined. The table names users by a figure number. The dashboards related to these usage patterns are in the figures 25 to 29 below (The data represented in the dashboards has been replaced with dummy data). The usage data of these users has been extracted from the activity log. Combined with the structure data of the ERP system the pages the user visited can be linked to a submenu item. If a user visited five or more pages related to a submenu item, a 1 is linked to this submenu item. If a user visited less than 5 pages related to a submenu item, this submenu item gets a 0 linked to it.

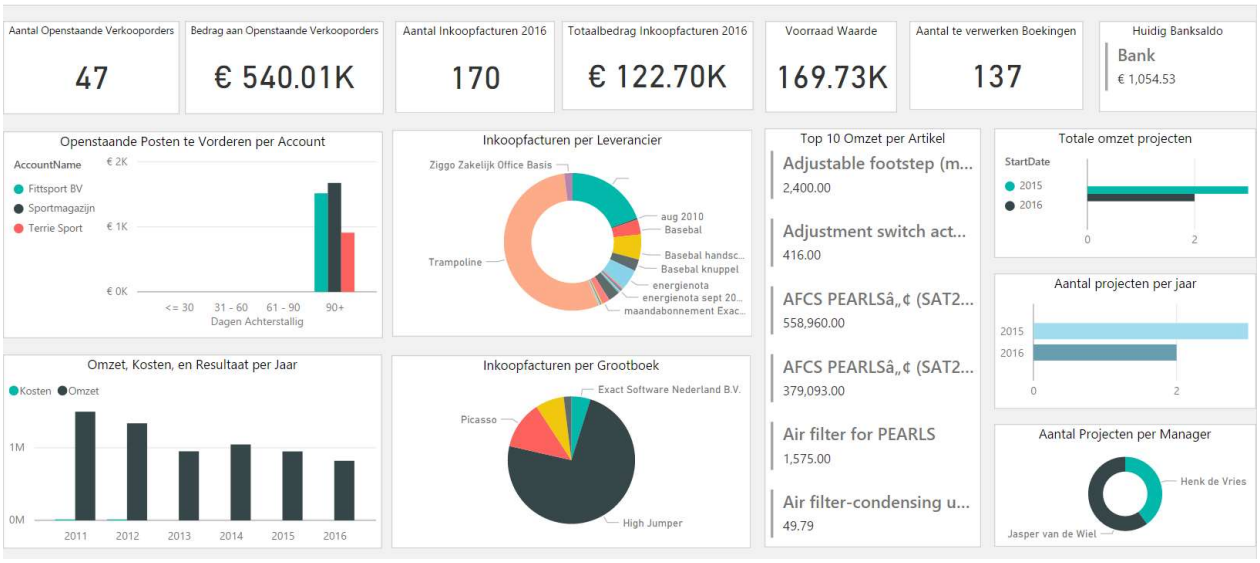

Figure 27. Dashboard for the second usage pattern shown in table 14

Table 14 shows that the user of for instance the dashboard in figure 26 has a usage pattern that is: “10000100000000010010”. The prototype can compare this string to the dashboard components. It will then show the components that are related to the submenu items that have a one linked to them, in the usage pattern, on the dashboard.

Table 14. Usage patterns the dashboards in respectively figure 26, 27, 28 and 29 are based on.

Submenu Item	Figure 26	Figure 27	Figure 28	Figure 29
Accounts	1	1	1	1
Contact Persons	0	0	0	0
Quotations	0	0	0	0
Bank	0	1	1	0
Currency	0	0	0	0
Entries	1	1	1	0
G/L Accounts	0	1	1	0
Employees	0	0	0	0
Payroll	0	0	0	0
Entry	0	0	0	0
Invoices	0	0	1	0
Projects	0	1	0	0
Planning	0	1	0	0
Inventory	0	1	1	0
Purchase Orders	1	1	0	1
Returns	0	0	0	1
Items	0	1	1	1
Sales Orders	1	1	1	0
Subscriptions	0	0	0	0

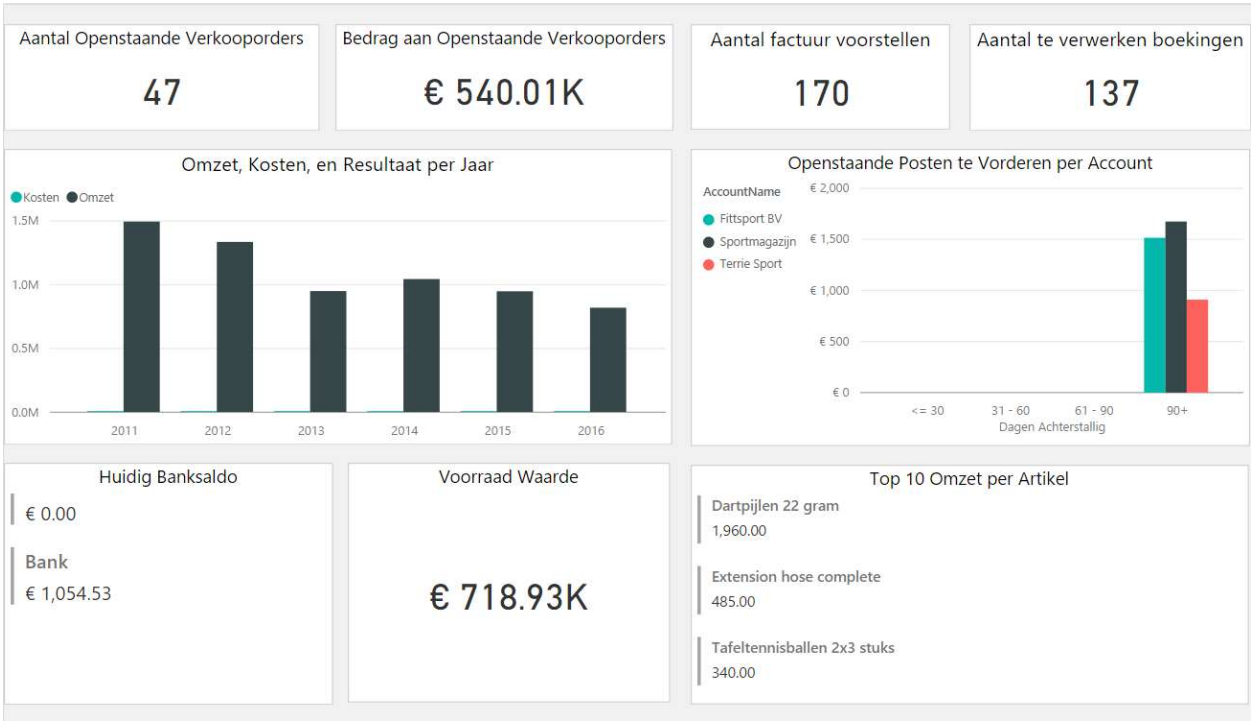

Figure 28. Dashboard for the third usage pattern shown in table 14.

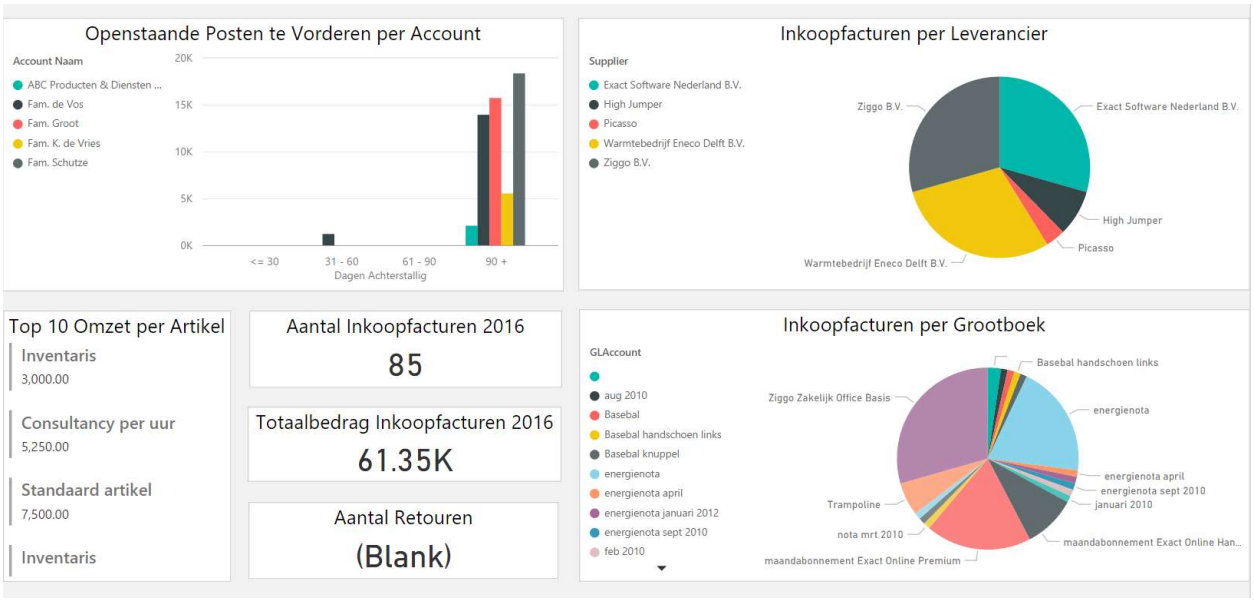

Figure 29. Dashboard for the fourth usage pattern shown in table 14.

9. Evaluation

For the evaluation of the model, a case study has been conducted. The case study company is Exact Software Nederland B.V. (From now on “Exact”). Exact is a business software provider with its headquarters in Delft, The Netherlands. Exact has around 1550 employees based in 15 countries. They serve over 250.000 businesses in 110 countries with their business software. Exact provides both on premise and SaaS products. Since this research is focused on SaaS, the case study is focused on exacts’ SaaS ERP solution called “Exact Online” (Later referred to as EOL). Exact online is a SaaS Multi-Tenant ERP product targeted at SMEs and accountants.

9.1 Case Study Protocol

In the case study, two exercises are employed. The first exercise makes use of a paper prototype, and the second exercise makes use of the developed dashboard generator prototype.

In exercise 1, the case study participant is provided with the 28 components a dashboard can consist of. These components are provided to the participant printed out on paper, containing demo data. The participant is asked to pick their top N preferred components they would like to see in their ideal personal dashboard. The number N is determined based on how many components the model suggests are in the ideal dashboard of that specific user.

After this exercise, the participant is shown the dashboard generated by the dashboard generator based on their user type. This dashboard contains the real user data. Now, the differences between the participants’ picks and the dashboard generated by the prototype are observed. If there are any differences these are discussed. Does the participant still like his/her own selection better than the generators choice? Or did the participant change his/her mind now he/she sees the components with real company data. What is the reason the user does not like the components suggested by the dashboard generator?

After these two exercises, the participant is asked if he/she would have selected more or fewer components than the number they have been asked to select. If this number is lower, the participants are asked to discard the components they would not have picked without the mandatory. Moreover, if this number is higher, the user is asked to name the components they would have liked to include as well. And finally the participant is asked if they miss any components in the list of 28 components they can choose from.

9.2 Case Study Participants

In the case study the prototype is evaluated with 15 participants. All participants are users of the same SaaS ERP system. The number of case study participants beforehand was not predetermined. After twelve evaluation sessions the theoretical saturation, as described by Glaser and Strauss (1967) was achieved. This means the point is reached at which no new information is added to the findings. However, at this point five manufacturing users, five production users, and only two professional services users participated in the case study. Therefore, the decision has been made to speak to three more professional services users in order to have an even distribution between the three packages.

10. Results

In this chapter, the findings of the case study have been documented. Fifteen evaluation sessions have been conducted with customers of the case study company. The case study results answer sub research question 4:

> ***SRQ4**: To what extent does a dashboard, generated based on a model of different usage patterns, satisfy the information needs of individual SME employees?*

The findings are documented per package first, next in the sub-conclusion of this chapter the results of all participants together are summarized (The results of the separate cases can be found in appendix C). Three packages have been studied in this research, and for each package, five case study subjects have conducted the case study assignments. Because of confidentiality, their subscription type, combined with a number, is used in order to distinguish the case study subjects. Per package, the percentage of correct suggestions of components, and subcategories, is given in a table. The distinction between components and subcategories is made because some submenu items are linked to more than one component. This means, if the submenu item prescribes two components for a subcomponent the percentage of correct components is lower if a user only selects one of those two components. Whereas, the percentage of correct suggestions of subcategories is not influenced by this. This distinction enables us to test two phases of our research separately, namely the pattern discovery phase and the pattern analysis phase. The percentage of correct suggestions of subcategories tests the model delivered after pattern discovery. The percentage of correct suggestions of components tests the information needs linked to these subcategories in the pattern analysis phase. Because these percentages alone are not to answer the research question, the case study participants' opinions are taken into account. For this reason, some statements made during the evaluation sessions are included in the text below. Finally, in the conclusion of this chapter sub research question 4 is answered.

In appendix C, each case study evaluation is discussed separately. For each subject a description of what the user uses exact for, if the user wanted to pick more or fewer components than prescribed by the model, and if there are any components they would like to see on a dashboard that are not in the 28 suggested components, are given. Finally, interesting remarks following from the discussion on the differences between the users' picks and the suggestion by the model are described.

10.1 Case Study Results per Package

10.1.1 Manufacturing

Table 15. Case study results of the five manufacturing users

User #	% of correct components	% of correct subcategories
1	80.00	83.33
2	25.00	100.00
3	83.33	80.00
4	86.67	80.00
5	80.00	88.89
Average	71.00	86.44

The users with a manufacturing subscription, in general have a company that produces goods. This can for instance be hospital beds, or cutting dies. As can be seen in table 15 above, the average correct component suggestion for the five case study

participants is 71.00%. The percentage of correct subcategories is higher, at 86.44%. User number 2 causes the substantially lower percentage of correct component suggestion.

"This dashboard is not perfect but it's a good start" – Manufacturing user 3

10.1.2 Wholesale

Table 16. Case study results of the five wholesale users

User #	% of correct components	% of correct subcategories
1	81.82	71.43
2	80.00	90.91
3	66.67	75.00
4	87.50	85.71
5	88.89	87.50
Average	80.98	82.11

The users with a wholesale subscription are in general purchasing, selling and distributing products as an intermediary. This means most wholesale users are working at a business-to-business company. Of the five participants, this holds true for three users. The two others are respectively a company that sells services, and a company with a web shop.

"This dashboard is more relevant to me than the standard dashboards" – Wholesale user 2

As can be seen in table 16 above, the percentage of correct component suggestion is 80.98 %, and the percentage of correct submenu item suggestion is only slightly higher at 82.11%.

"We use the standard dashboards a lot, but a personalized dashboard like this one would be appreciated" - Wholesale user 4

10.1.3 Professional Services

Table 17. Case study results of the five professional services users

User #	% of correct components	% of correct subcategories
1	77.78	83.33
2	76.92	100.00
3	84.62	87.50
4	85.71	88.89
5	83.33	100.00
Average	81.67	91.94

The users of the professional services packages are not as easily generalizable as the previous two packages. Users can have a business in a large range of disciplines. Users can for instance have an advertising agency, a consultancy firm or can be an IT services provider. One of our case study participants runs an advertising agency, and another one runs a design agency.

"I already have these components in my custom dashboard" – PSA User 1

From table 17 above. We see again that with an 81.67% correct component suggestion and a 91.94% correct subcategory suggestion that the correct subcategory suggestion is higher than the correct component suggestion.

10.2 Chapter Conclusion

In order to answer the fourth research question "To what extent does a dashboard, generated based on usage patterns, satisfy the information needs of individual SME employees?" the case study results of the 3 packages can be summarized into two average percentages. These percentages are shown below.

Table 18. Summarized case study results

Package	% of correct components	% of correct subcategories
Manufacturing	71.00	86.44
Wholesale	80.98	82.11
PSA	81.67	91.94
Average	77.88	86.83

As can be seen the average correct component suggestion is at 77.88%, and the average correct submenu item suggestion is at 86.83%. This suggests that the usage patterns are more reliable than the information needs/components linked to these usage patterns. As to answer the research question, the answer can be expressed in the form of these average percentages. More textual conclusions can be deducted from these percentages. Because the results are not 100% accurate, it can be argued that a dashboard, generated based on usage patterns, does not satisfy the information needs of users. An argument for this would be that on average around 1/4th of the components is not suggested correctly. On the other hand, it can be argued that the results are satisfactory because 3/4th of the components is suggested correctly. These opposing statements show that these conclusions are subject to opinion. In order to decide which of the two arguments is most likely true, the opinion of the case study participants can be taken into account.

From the general opinion during the evaluation sessions it can be concluded that the dashboard generator based on usage patterns satisfies the users' information needs more than the standard dashboards they are provided with at the moment. Ten out of the 13 case study participants expressed a positive opinion about the generated dashboard. In general, the overview delivered by our prototype provided the participants with better information, or with equal information but with less noise and without having to change pages, or said that the dashboard was similar to the dashboard they composed themselves.

One of the case study participants mentioned that creating a custom dashboard, as already offered by EOL, has been on his to do list for a while. However, because of the effort this takes when starting from scratch, he has not yet done this. He mentions that if the dashboard generated by the prototype would show up in this section as a start, it would be less work, and he would be more inclined to customize the dashboard to fit his needs. Other case study participants have made similar statements.

Another opinion that has been voiced is that the dashboard shows the same components as the custom dashboard a participant has already created for himself in EOL. This means the generated dashboard does not provide any benefit for this particular participant. However, if this dashboard had shown when he started creating his custom dashboard it would have costed him less time.

If the case study participants' opinions are considered, the conclusion is that the dashboards are satisfactory. In general, the case study participants are happy with more than 3/4th of the components on the suggested dashboard, and the statements made during the evaluation sessions endorse this.

11. Discussion & Future Work

11.1 Discussion

In this chapter, the limitations faced in this research are discussed. First, the internal validity, external validity, and reliability are discussed. Next, other limitations not regarding these three concepts are discussed.

To ensure the internal validity of this research, some steps are taken to limit the chance of errors in the data. All steps for the model development have used the exact same dataset. All evaluation sessions have been conducted using the same interview protocol, and by using the same prototype, components, and dataset.

External validity is related to the ability to generalize this research to subjects other than the case study used in this research. The fact that this research has been conducted using only the data of a single case study company, means some company specific decisions have been made. For instance, focusing on wholesale, production and professional services customers. However, the main contribution of this research is a method that enables a company to specify types of users in their customer base, and generating dashboards based on these user types. This method assumes that a company has an activity log in their product that stores separate user actions. Moreover, the method assumes that there is extra information available to enhance this activity log. Given the fact that almost all ERP systems meet these requirements the method can be generalized to other ERP software companies.

Reliability is related to the ability to repeat this research with the same results. This means other researchers must be able to replicate the steps taken in this research. And by doing so arrive at the same conclusions. For the first part of the research, this is trivial, since the steps from data gathering, to data pre-processing, to pattern discovery, to pattern analysis have been explained in detail. And since real company data has been used, and software that is available for free, in this research the research can be replicated identically. The same goes for the development of the prototype. This leaves us with the question if the opinion of 15 case study participants represents the opinion of all customers. In our opinion 15 case study participants give a good representation of the customer base, since the results from each separate customer interview are similar. This satisfies the theoretical saturation criterion as explained in chapter 9.

After defending the steps taken in this research, we feel the need to discuss some limitations that can be improved in future research. For starters, the activity log at our case study company is not the most detailed activity log possible. The user actions are expressed by aspx requests. This means it is known what aspx a user requests. However, this does not express what the user is doing exactly on this aspx. Certain aspx'es have more than one functionality depending on the parameters given with it in the request. Another limitation in the activity log is that a user can conduct multiple actions on a single aspx without this showing up in the activity log. In future research a case study company with a more detailed activity log could improve the research.

Another limitation on the data at our case study company is the data available for enhancing the activity log. An example of this is the information about company size. This data has been studied in order to find possibilities to enhance the dataset. Unfortunately, it turned out the data about company size was not available for the majority of the customer companies, and if it was available it often turned out not to be reliable. If this data had been available, company size could have played a role in the selection of dashboard components. For instance, not picking the "Projects per Manager" component for companies with 1-5 employees.

11.2 Future work

The results of this research show that a model based on usage and structure data of an ERP system can provide a basis for personalized dashboards. These results provide multiple insights for further research.

The first insight, that follows directly from the difference between the percentage of correct component suggestions and correct subcategory suggestions, is that more research should be performed on the linking of dashboard components to usage patterns. The percentages show that in the prototype subcategory suggestion is better than component suggestion. This is partially explainable by the fact that parameters are not logged in the dataset used for this research, with a more detailed activity log there would be more information available in order to base the component selection on. Another idea for future research is improving the current method and prototype to a point where the successful suggestion percentage reaches 100%. This can be achieved by completing more cycles in the design science process.

Further research could also be performed by not only using the proposed method for the generation of personalized dashboards. With this in mind model-driven development is a field of research that can benefit from the application of usage mining. The method could for instance be extended with the goal of tailoring other components of a SaaS ERP system, or even tailoring the whole system based on usage patterns. Another way of extending the model is not extending the goal of the method, but extending the method to other products. In this research the data of only one SaaS ERP system has been used for the analysis. It is interesting to study how the method performs in for instance another SaaS ERP system. Or maybe even another kind of product that also employs dashboards.

Another way to extend this research, is applying other data mining techniques. In this research a number of techniques have been used in order to discover different usage patterns. The techniques of process mining and FCA in this research have been selected based on their perceived usefulness in dashboard personalization. However, a number of techniques have not been looked into in much detail. For instance, for sequential mining only process mining has been studied in detail. Pachidi et al. (2014) use markov chains in order to discover interesting usage paths in an ERP system.

12. Conclusion

In the first chapter of this research, the main research question has been formulated based on the problem statement. The goal of this research is answering this question. In order to do so, this research has focused on using web usage mining in order to provide user specific BI dashboards to ERP system users. Web usage mining has been used to find different usage patterns in an online SaaS ERP system. Information needs have been linked to these usage patterns, and based on these patterns and information needs, dashboards have been generated that try to satisfy these information needs. The main research question formulated from the problem statement is the following:

> ***RQ:** How can a model of user information needs be developed from ERP usage statistics in order to generate user specific BI dashboards?*

The answer to this research question, and thus the conclusion of this research is: A model of user information needs can be developed from ERP usage statistics by performing web usage mining, applying Formal Concept Analysis as a clustering technique, on a dataset that consists of usage data and structure data. This research also shows that the model can be applied in order to generate user specific BI dashboards. The steps that need to be followed in order to develop a model of usage patterns, and applying this model in order to generate user specific BI dashboard have been expressed in the form of a PDD, the PDD can be found in figure 23 in chapter 7 of this document.

This conclusion follows from the answers to the four sub research questions that have been formulated in chapter 1. The sub research questions each answer part of the main research question. The answers to the sub research questions have been described in the chapters in which each sub research question has been addressed. Below the answers to each of the four research questions that have contributed to the conclusion of this research are summarized.

How can the different usage patterns of SME employees while using ERP systems be identified?

In order to answer this research question, and thus finding a way to identify unique usage patterns of SME employees using ERP systems, first extensive literature research has been performed. The literature research resulted in the conclusion that web usage mining technology has many intersections with the data that is available at ERP software vendors, and with the goal of our research. In web usage mining literature, multiple techniques are found in order to perform web usage mining. In all techniques, the first two steps are data gathering and data pre-processing. These two steps have been executed first. The next step in web usage mining is pattern discovery. A number of pattern discovery techniques has been studied and tried out. Examples of these techniques are association rule mining, process mining, and clustering. These techniques did not succeed in showing us different usage patterns in the data. Eventually we succeeded in identifying unique usage patterns by using Formal Concept Analysis, which is a clustering technique. We did however not succeed by using FCA on only the usage data. This data was too complex and has a low information density. This is why the usage data is combined with structure data. The structure data consist of the menu structure of the ERP system developed by our case study company.

The structure data enables us to group individual page views together in categories based on the menu structure. Page views can be grouped on main-menu item, or more specific on submenu item. By applying FCA on the page views grouped by main-menu item, clusters of usage patterns can be identified. Applying FCA on the page views grouped by submenu item provides a more

detailed view of different clusters of usage patterns. The finding of these patterns proves that usage patterns can be identified by using FCA. In short the answer to first sub-research question 'How can the different usage patterns of SME employees while using ERP systems be identified?' is: By using web usage mining on usage and structure data, employing Formal Concept Analysis as the pattern discovery technique.

What are the different usage patterns of SME employees while using ERP systems?

In order to express the unique usage patterns of SME employees while using ERP systems, and thereby answering the second sub research question formal concept analysis has been applied on the usage data of customers of our case study company. In order to do this, a dataset containing all user actions over the timespan of a month is combined with the structure data expressed by the menu structure of the ERP system. This led to usage patterns on different abstraction layers. The highest level of abstraction is that of menu items. This abstraction level reveals ~100 different usage patterns. Diving one level deeper to the submenu item abstraction level reveals ~3000 usage patterns. Looking at the lowest level of abstraction would be looking at single page views. This is computationally unfeasible and therefore the usage patterns are based on the submenu item abstraction level.

In order to create a model that can be used by others the findings have are described in the form of a simple recipe. In order to find the usage pattern of a user of an ERP system the following steps can be followed:

- Retrieve the menu structure of the ERP system;
- Gather the different pages of the ERP system;
- Clean this data in order to keep only the relevant page views;
- Link the relevant page views to their submenu items in order to form a page-keyword table;
- Retrieve the usage data for the user over a time span of 30 days;
- Link the usage data to the page-keyword table
- If a submenu item appears in the resulting dataset more than 5 times this submenu item is in the usage pattern of the user;
- The combination of submenu items from the above step expresses the usage pattern of this specific user.

These steps have been visualized in a PDD in order to create a clear overview of the steps and their resulting deliverables. These usage patterns are very general, and not yet usable for dashboard generation. In order to generate dashboards based on these usage patterns, information needs have been linked to the submenu items. This process is discussed in the answer to the next research question below.

How do the usage patterns express different information needs of SME employees while using ERP systems?

The information needs of different users are hypothetically expressed by the information that is on the pages they visit. In this research, the information on the different pages of the ERP system has been gathered by scraping the title elements of these pages. The information on these pages has been linked to dashboard components inspired by existing ERP dashboards, and these components have been evaluated with experts in the field. Linking these dashboard components to the usage data of the ERP system answers the third sub research question:"How do the usage patterns express different information needs of SME employees while using ERP systems?" Each usage pattern expresses different information needs of users of the ERP system based on their ERP system usage. The additional steps performed in order to link information needs to usage patterns have been added to the PDD created in answering the previous sub research question. The expanded PDD can be found in chapter 7, in figure 24. The last step in the research is evaluating the model, and the components linked to the different user types in the model.

To what extent does a dashboard, generated based on usage patterns, satisfy the information needs of individual SME employees?

In order to answer this question a prototype of a dashboard generator, which generates dashboards based on the usage patterns expressed by our model has been developed. With this prototype, a case study with 15 participants has been conducted. These participants are users of the ERP system developed by our case study company. Five participants of each package are studied in this research. On average, the model performed best on component suggestion for the professional services package, with a correct suggestion percentage of 81.66%. In addition, on the suggestion of correct submenu items, the model also performed best for the PSA package with a correct suggestion percentage of 91.94%. The average results per package as shown in table 19 below.

Table 19. Summarized case study results

Package	% of correct components	% of correct subcategories
Manufacturing	71.00	86.44
Wholesale	80.98	82.11
PSA	81.67	91.94
Average	77.88	86.83

To answer the research question, to what extent the dashboards satisfies the information needs of individual SME employees in another way than showing the percentages would not be an objective answer. Therefore, the answer to the research question is: The dashboard generated based on usage patterns on average satisfies 77.88% of the user's information needs. For the answer, the correct component suggestion percentage is selected; because this percentage is more related to the actual dashboard the user is confronted with than the percentage of correct submenu item suggestion.

References

Van Der Aalst, W., Adriansyah, A., De Medeiros, A. K. A., Arcieri, F., Baier, T., Blickle, T., ... & Burattin, A. (2011, August). Process mining manifesto. In *International Conference on Business Process Management* (pp. 169-194). Springer Berlin Heidelberg.

Adams, L., & Courtney, J. F. (2004, January). Achieving relevance in IS research via the DAGS framework. In *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on* (pp. 10-pp). IEEE.

Aldridge, G. (2008). Why 70% of Dashboard Initiatives Fail. (White paper)

Aulbach, S., Grust, T., Jacobs, D., Kemper, A., & Rittinger, J. (2008, June). Multi-tenant databases for software as a service: schema-mapping techniques. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data* (pp. 1195-1206). ACM.

Bass, L. (2007). *Software architecture in practice*. Pearson Education India.

Belohlavek, R. (2011, June). What is a Fuzzy Concept Lattice? II⋆. In Rough Sets, Fuzzy Sets, Data mining and Granular Computing: 13th International Conference, RSFDGrC 2011, Moscow, Russia, June 25-27, 2011, Proceedings (Vol. 6743, p. 19). Springer Science & Business Media.

Bernroider, E., & Koch, S. (2001). ERP selection process in midsize and large organizations. *Business Process Management Journal*, *7*(3), 251-257.

Bezemer, C. P., Zaidman, A., Platzbeecker, B., Hurkmans, T., & Hart, A. (2010a, September). Enabling multi-tenancy: An industrial experience report. In *Software Maintenance (ICSM), 2010 IEEE International Conference on* (pp. 1-8). IEEE.

Bezemer, C. P., & Zaidman, A. (2010b, September). Multi-tenant SaaS applications: maintenance dream or nightmare? In *Proceedings of the Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE)* (pp. 88-92). ACM.

Bezemer, C. (2014). *Performance Optimization of Multi-Tenant Software Systems*. TU Delft, Delft University of Technology.

Buxmann, P., Hess, T., & Lehmann, D. W. I. S. (2008). Software as a Service. W*irtschaftsinformatik*, *50*(6),500-503.

Cheng, B. H., & Atlee, J. M. (2007, May). Research directions in requirements engineering. In *2007 Future of Software Engineering* (pp. 285-303). IEEE Computer Society.

Chong, F., & Carraro, G. (2006). Architecture strategies for catching the long tail. *MSDN Library, Microsoft Corporation*, 9-10.

Chou, D. C., Bindu Tripuramallu, H., & Chou, A. Y. (2005). BI and ERP integration. *Information Management & Computer Security*, *13*(5), 340-349.

Cimiano, P., Hotho, A., Stumme, G., & Tane, J. (2004). Conceptual knowledge processing with formal concept analysis and ontologies. In *Concept Lattices*(pp. 189-207). Springer Berlin Heidelberg.

Cimiano, P., Hotho, A., & Staab, S. (2005). Learning Concept Hierarchies from Text Corpora using Formal Concept Analysis. *J. Artif. Intell. Res.(JAIR)*, *24*, 305-339.

D'souza, Austin; Kabbedijk, Jaap; Seo, DongBack; Jansen, Slinger; and Brinkkemper, Sjaak, "Software-As-A-Service: Implications For Business And Technology In Product Software Companies" (2012). PACIS 2012 Proceedings. Paper 140.

Dai, H., & Mobasher, B. (2003, June). A Road Map to More Effective Web Personalization: Integrating Domain Knowledge with Web Usage Mining. In *International Conference on Internet Computing* (pp. 58-64).

Van Dongen, B. F., de Medeiros, A. K. A., Verbeek, H. M. W., Weijters, A. J. M. M., & Van Der Aalst, W. M. (2005, June). The ProM framework: A new era in process mining tool support. In *International Conference on Application and Theory of Petri Nets* (pp. 444-454). Springer Berlin Heidelberg.

Dubey, A., & Wagle, D. (2007). Delivering software as a service. *The McKinsey Quarterly*, *6*(2007), 2007.

Eckerson, W. (2003). Smart companies in the 21st century: The secrets of creating successful business intelligence solutions. *TDWI Report Series*, *7*.

Eckerson, W. (2010). *Performance dashboards: measuring, monitoring, and managing your business*. John Wiley & Sons.

Eekels, J., & Roozenburg, N. F. (1991). A methodological comparison of the structures of scientific research and engineering design: their similarities and differences. *Design Studies*, *12*(4), 197- 203.

Engebrethson, R. (2012). Comparative Analysis of ERP Emerging Technologies.

Van Everdingen, Y., Van Hillegersberg, J., & Waarts, E. (2000). Enterprise resource planning: ERP adoption by European midsize companies. *Communications of the ACM*, *43*(4), 27-31.

Evelson, B. (2010). The Forrester Wave™: Enterprise Business Intelligence Platforms, Q4 2010. *Intelligence (BI)*, *3*.

Fayyad, U. M., Piatetsky-Shapiro, G., & Smyth, P. (1996, August). Knowledge discovery and data mining: towards a unifying framework. In *KDD* (Vol. 96, pp. 82-88).

Fensel, D. Ontologies: A silver bullet for knowledge management and electronic-commerce (2000). *Berlin: Spring-Verlag*.

Few, S. (2006). *Information dashboard design* (pp. 120-206). O'Reilly.

Formica, A. (2006). Ontology-based concept similarity in formal concept analysis. *Information Sciences*, *176*(18), 2624-2641.

Fui-Hoon Nah, F., Lee-Shang Lau, J., & Kuang, J. (2001). Critical factors for successful implementation of enterprise systems. *Business process management journal*, *7*(3), 285-296.

Frank, U., Heise, D., & Kattenstroth, H. (2009). Use of a domain specific modeling language for realizing versatile dashboards. In *Proceedings of the 9th OOPSLA workshop on domain-specific modeling (DSM)*.

Ganter, B., Stumme, G., & Wille, R. (2002). Formal concept analysis: Methods and applications in computerscience. *TU Dresden, http://www. aifb. uni-karlsruhe. de/WBS/gst/FBA03. shtml*.

Ganter, B., & Wille, R. (2012). *Formal concept analysis: mathematical foundations*. Springer Science & Business Media.

Gile, K. (2006). The Forrester Wave™: BI Reporting And Analysis Platforms, Q1 2006. *Forrester, Fevereiro*.

Glaser, B. G., & Strauss, A. L. (1998). Grounded theory. *Strategien qualitativer Forschung. Bern: Huber*.

Guarino, N. (1998). *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy* (Vol. 46). IOS press.

Guo, C. J., Sun, W., Jiang, Z. B., Huang, Y., Gao, B., & Wang, Z. H. (2011). Study of software as a service support platform for small and medium businesses. In *New Frontiers in Information and Software as Services* (pp. 1-30). Springer Berlin Heidelberg.

Griffin, J. (2007). BI and ERP integration: Five critical questions. *Information Management*, *17*(5), 6.

Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing?. *International journal of human-computer studies*, *43*(5), 907-928.

Haddara, M., & Zach, O. (2011, January). ERP systems in SMEs: A literature review. In *System Sciences (HICSS), 2011 44th Hawaii International Conference on* (pp. 1-10). IEEE.

Hevner, March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, *28*(1), 75–105.

Jansen, S., Houben, G. J., & Brinkkemper, S. (2010). Customization realization in multi-tenant web applications: case studies from the library sector (pp. 445-459). Springer Berlin Heidelberg.

Jones, D. (2009). *The Shorcut Guide to Achieving Business Intelligence in Midsize Companies*.Realtimepublishers.com.

Kabbedijk, J., & Jansen, S. (2011). Variability in multi-tenant environments: architectural design patterns from industry. In *Advances in conceptual modeling. Recent developments and new directions* (pp. 151-160). Springer Berlin Heidelberg.

Kabbedijk, J., Pors, M., Jansen, S., & Brinkkemper, S. (2014). Multi-tenant Architecture Comparison. In *Software Architecture* (pp. 202-209). Springer International Publishing.

Kaplan, M.J. 2005. SaaS Survey Shows New Model Becoming Mainstream. Cutter Consortium Executive Update, 6(22), 1-5.

Kaplan, J. M. (2007). Saas: Friend or foe?. Business Communications Review, 37(6), 48.

Kawamoto, T., & Mathers, B. (2007). Key success factors for a performance dashboard. *Information Management*, *17*(7), 20.

Kumar, K., & van Hillegersberg, J. (2000). Enterprise resource planning: Introduction. *Communications of the ACM*, *43*(4), 22-26.

Kwok, T., T. Nguyen, and L. Lam (2008). "A software as a service with multi-tenancy support for an electronic contract management application". In: Proceedings of the International Conference on Services Computing (SCC). IEEE, pp. 179–186.

Lidwell, W., Holden, K., & Butler, J. (2010). *Universal principles of design, revised and updated: 125 ways to enhance usability, influence perception, increase appeal, make better design decisions, and teach through design*. Rockport Pub.

Liu, B. (2007). Web usage mining. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*, 449-483.

Mäkilä, T., Järvi, A., Rönkkö, M., & Nissilä, J. (2010). How to Define Software-as-a-Service–An Empirical Study of Finnish SaaS Providers. In *Software Business* (pp. 115-124). Springer Berlin Heidelberg.

Makkar, G. D., & Bist, M. (2012). EaaS-ERP as a Service. *Journal of Information and Operations Management*, *3*(1), 141.

Mertz, S. A., Eschinger, C., Eid, T., & Pring, B. (2007). Dataquest insight: SaaS demand set to outpace enterprise application software market growth. *Gartner report*.

Mobasher, B., Cooley, R., & Srivastava, J. (2000). Automatic personalization based on web usage mining. *Communications of the ACM*, *43*(8), 142-151.

Mobasher, B. (2007). Data mining for web personalization. In *The adaptive web* (pp. 90-135). Springer Berlin Heidelberg.

Nitu (2009). Configurability in SaaS (software as a service) applications. In *Proc. 2nd Annual Conference on India Software Engineering Conference* (pp. 19-26).

Nunamaker Jr, J. F., & Chen, M. (1990, January). Systems development in information systems research. In *System Sciences, 1990., Proceedings of the Twenty-Third Annual Hawaii International Conference on* (Vol. 3, pp. 631-640). IEEE.

Nuseibeh, B., & Easterbrook, S. (2000, May). Requirements engineering: a roadmap. In *Proceedings ofthe Conference on the Future of Software Engineering* (pp. 35-46). ACM.

Obitko, M., Snasel, V., Smid, J., & Snasel, V. (2004, September). Ontology Design with Formal Concept Analysis. In *CLA* (Vol. 110).

Öhgren, A., & Sandkuhl, K. (2005, February). Towards a methodology for ontology development in smalland medium-sized enterprises. In *IADIS AC* (pp. 369-376).

Pachidi, S., Spruit, M., & Van De Weerd, I. (2014). Understanding users' behavior with software operation data mining. *Computers in Human Behavior*, *30*, 583-594.

Palpanas, T., Chowdhary, P., Mihaila, G., & Pinel, F. (2007). Integrated model-driven dashboard development. *Information Systems Frontiers*, *9*(2-3), 195-208.

Papazoglou, M. P. (2003, December). Service-oriented computing: Concepts, characteristics and directions.In *Web Information Systems Engineering, 2003. WISE 2003. Proceedings of the Fourth International Conference on* (pp. 3-12). IEEE.

Peffers, K., Tuunanen, T., Gengler, C. E., Rossi, M., Hui, W., Virtanen, V., & Bragge, J. (2006, February). The design science research process: a model for producing and presenting information systems research. In *Proceedings of the first international conference on design science research in information systems and technology (DESRIST 2006)* (pp. 83-106).

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research.*Journal of management information systems*, *24*(3), 45-77.

Priss, U. (2006). Formal concept analysis in information science. *Arist*, *40*(1), 521-543.

van Rees, R. (2003). Clarity in the usage of the terms ontology, taxonomy and classification. *CIB REPORT*, *284*(432), 1-8.

Rossi, M., & Sein, M. K. (2003). Design research workshop: a proactive research approach. *Presentation delivered at IRIS*, *26*, 9-12.

Sharda, R., Aronson, J. E., & King, D. (2008). *Business intelligence: A managerial approach*. Upper Saddle River: Pearson Prentice Hall.

Srivastava, J., Cooley, R., Deshpande, M., & Tan, P. N. (2000). Web usage mining: Discovery and applications of usage patterns from web data. *Acm Sigkdd Explorations Newsletter*, *1*(2), 12-23.

Steinebach, G., Guhathakurta, S., & Hagen, H. (Eds.). (2009). *Visualizing sustainable planning*. Springer.

Stumme, G. (2009). Formal concept analysis. In *Handbook on Ontologies* (pp. 177-199). Springer Berlin Heidelberg.

Takeda, H., Veerkamp, P., & Yoshikawa, H. (1990). Modeling design process. *AI magazine*, *11*(4), 37.

Thompson, O. (2004). Business Intelligence Success, Lessons Learned. Technology.

Tutunea, M. F., & Rus, R. V. (2012). Business intelligence solutions for SME's. *Procedia Economics and Finance*, *3*, 865-870.

Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, methods and applications. *Knowledge engineering review*, *11*(2), 93-136.

Walls, J. G., Widmeyer, G. R., & El Sawy, O. A. (1992). Building an information system design theory for vigilant EIS. *Information systems research*, *3*(1), 36-59.

Wang, Z. H., Guo, C. J., Gao, B., Sun, W., Zhang, Z., & An, W. H. (2008, October). A study and performance evaluation of the multi-tenant data tier design patterns for service oriented computing. In *e-Business Engineering, 2008. ICEBE'08. IEEE International Conference on* (pp. 94-101). IEEE.

Wang, L., Von Laszewski, G., Younge, A., He, X., Kunze, M., Tao, J., & Fu, C. (2010). Cloud computing: a perspective study. *New Generation Computing*, *28*(2), 137-146.

Watson, H. J., Wixom, B. H., Hoffer, J. A., Anderson-Lehman, R., & Reynolds, A. M. (2006). Real-time business intelligence: Best practices at Continental Airlines. *Information Systems Management*, *23*(1), 7.

Watson, H. J., & Wixom, B. H. (2007). The current state of business intelligence. *Computer*, *40*(9), 96-99.

Wieringa, R. (2009, May). Design science as nested problem solving. In Proceedings of the 4th international conference on design science research in information systems and technology (p. 8). ACM.

Wixom, B., & Watson, H. (2012). The BI-based organization. Organizational Applications of Business Intelligence Management: Emerging Trends, IGI Global, Hershey, 193-208.

Xu, L. D. (2011). Enterprise systems: state-of-the-art and future trends. *Industrial Informatics, IEEE Transactions on*, *7*(4), 630-640.

Appendices

Appendix A. Aspx'es, Related (Sub)Menu Items, and Title Elements

Category	Subcategory	Aspx	Title element
Accountancy	General	AccBatchReports.aspx	Management|Batch reports
Accountancy	General	AccCompareFinPeriods.aspx	Compare financial periods
Accountancy	Management	AccCustomerManagement.aspx	Management|Master data
Accountancy	General	AccGLJournalOverview.aspx	G/L Account - Journal
Accountancy	Management	AccLiquidAssets.aspx	Liquidity overview per company
Accountancy	Management	AccProcessManagement.aspx	Financial process management
Accountancy	Management	AccTransManagement.aspx	Management|Transactions
Accountancy	Management	AccVATManagement.aspx	Management: VAT return
Financial	Taxes and returns	AccVATMatching.aspx	VAT-matching
Financial	Assets	AssAssetList.aspx	Overview: Assets
Financial	Assets	AssAssetStatements.aspx	Assets statement
Financial	Assets	AssAssetTransactions.aspx	Transactions: Asset
Financial	Assets	AssBalanceList.aspx	Assets-Checklist
Financial	Assets	AssDeprPlanning.aspx	Deprecation Plan
Financial	Assets	AssHistorySheet.aspx	Assets History sheet
Financial	Assets	AssInvestmentDeduct.aspx	Invesetment - Deduction
Financial	Assets	AssInvestmentList.aspx	Investment - List
Financial	Bank	BOPspPayments.aspx	Payments
Financial	Bank	CflAllocationRules.aspx	Overview: Allocation rules
Financial	Bank	CflBankFiles.aspx	Bank files
Financial	Bank	CflBatchBookingOverview.aspx	Overview: Direct debit batches
Financial	Bank	CflCashflowTransactions.aspx	Overview outgoing cheques
CRM	Accounts	CflCMScenarioActions.aspx	Management: Credit
Financial	Bank	CflDDMandates.aspx	Overview: Mandates
Financial	Bank	CflFinancialStatements.aspx	Overview statements
Financial	Bank	CflFinancialStmtEntry.aspx	Enter Manual statement
Financial	Bank	CflImport.aspx	Import statements
Financial	Bank	CflLinks.aspx	Bank Links
Financial	Bank	CflPayments.aspx	Payments
Financial	Bank	CflPaymentsCreditCard.aspx	Select Creditcard Payments
Financial	Bank	CflPaymentsNew.aspx	Pay by cheque
Financial	Bank	CflRemittances.aspx	Remittance adivce
Financial	Bank	CflStatements.aspx	To be reconciled
CRM	Campaigns	CMPCampaign.aspx	Campagin
CRM	Campaigns	CMPCampaigns.aspx	Overview: Campaigns
CRM	Accounts	CRMACcountNew.aspx	Create: Account
CRM	Accounts	CRMAccounts.aspx	Overview: Accounts
CRM	Accounts	CRMAccountSearch.aspx	Search: Accounts

CRM	Accounts	CRMAccountsimport.aspx	Import accounts
CRM	Accounts	CRMACcountValidation.aspx	Accounts: Validation
CRM	Accounts	CRMAccountValList.aspx	Accounts: Validation
CRM	Contact persons	CrmContact.aspx	Contact
CRM	Accounts	CRMContacts.aspx	Overview: Contacts
CRM	Accounts	CRMContactSearch.aspx	Search: Contacts
Financial	Bank	CrmSepaCustomerBankAccounts.aspx	Debtor: Bank accounts
CRM	Accounts	FinARAPAging.aspx	Ageing analysis: A/R
CRM	Accounts	FinARAPBalance.aspx	Balance list: A/R
CRM	Accounts	FinARAPOutstanding.aspx	Outstanding items: A/R
CRM	Accounts	FINARAPReminders.aspx	Print Reminders
Financial	G/L Accounts	FinBalanceSheet.aspx	Balance sheet / Profit and loss statement
Financial	Bank	FinBankEntities.aspx	Overview|Bank accounts
Financial	Budgets	FinBudgets.aspx	Budgets
Financial	Budgets	FinBudgetScenarios.aspx	Overview: Budget scenarios
Financial	Entries	FinFixedEntries.aspx	Overview: Recurring entries
Financial	G/L Accounts	FinGLAccount.aspx	Overview: G/L Accounts
Financial	Taxes and returns	FinIntrastatReturns.aspx	Overview: INTRASTAT returns
Sales	Sales Orders	FinInvoices.aspx	Print invoices
Financial	Entries	FinJournalSelect.aspx	Entry - New
Financial	Entries	FinMissingEntryNumbers.aspx	Overview: Missing entry numbers
Financial	G/L Accounts	FinPeriodDateCheck.aspx	Period-date check
Financial	G/L Accounts	FinProcessResults.aspx	Year end closing: Process profit / loss
Financial	Currency	FinRevaluation.aspx	Currency - Revaluation
Financial	G/L Accounts	FinTransactionSearch.aspx	Search: Financial - Transactions
Financial	Entries	FinUnprocessedEntries.aspx	Overview: Entries
Financial	Taxes and returns	FinVATCheckList.aspx	VAT checklist
Financial	Taxes and returns	FinVatReturns.aspx	VAT returns
Financial	Taxes and returns	FinVATSalesLists.aspx	EC sales list
Financial	Taxes and returns	FinVATTransactions.aspx	VAT overview
CRM	Quotations	GenProcessJobs.aspx	Print Quotations
Accountancy	General	GenTopicDefineTopics.aspx	Define template
Person	Payroll	HRMAbsencePercentages.aspx	Absence percentages: Overview
Person	Absence	HRMAbsenceRegs.aspx	Overview: Leave registrations
Financial	Cost centres	HRMCostCenters.aspx	Overview: Cost centres
Financial	Costs units	HRMCostUnits.aspx	Overview: Cost units
Person	Employees	HRMEmplAnniversaries.aspx	Overview: Anniversaries
Person	Employees	HRMEmplBirthdayList.aspx	Birthday list
Person	Employees	HRMEmplContracts.aspx	Overview: Employment contracts
Person	Employees	HRMEmployees.aspx	Overview: Employees
Person	Absence	HRMIllnessRegs.aspx	Overview: Absence registrations
Person	Employees	HRMPFEmployees.aspx	Overview|Pro forma employees
Person	Payroll	HRMSchedules.aspx	Overview: Schedules

Inventory	Planning	InvSalesForecasts.aspx	Sales forecast
Purchase	Inventory	InvStockCountNew.aspx	Stock count - New
Purchase	Inventory	InvStockCounts.aspx	Overview: Stock counts
Purchase	Inventory	InvStockPositions.aspx	Stock positions
Purchase	Inventory	InvStockValue.aspx	Stock value
Sales	Items	InvTransactions.aspx	Transaction list
Purchase	Warehouse transfers	InvWarehouseTransfer.aspx	Warehouse transfer
Purchase	Warehouse transfers	InvWarehouseTransferDeliveries.aspx	Print warehouse transfer
Purchase	Warehouse transfers	InvWarehouseTransfers.aspx	Overview: Warehouse transfers
Manufacturing	Realization	LicDivisionSettings.aspx	Company settings
Sales	Sales Orders	LogEntries.aspx	Overview: Sales invoices
Sales	Sales Orders	LogFixedEntries.aspx	Overview: Recurring sales invoices
Sales	Items	LogItem.aspx	Item
Purchase	Inventory	LogItemCostPriceChanges.aspx	Cost price changes
Sales	Items	LogItemGroups.aspx	Overview: Item groups
Sales	Items	LogItemLabelsPrint.aspx	Print item labels
Sales	Items	LogItemPrices.aspx	Overview: Item prices
Sales	Items	LogItemPriceUpdate.aspx	Update prices
Sales	Items	LogItems.aspx	Overview: Items
CRM	Marketing Lists	MarMarketingLists.aspx	Overview: Marketing lists
CRM	Marketing Lists	MarMarketingListWizard.aspx	New: Marketing list
Accountancy	General	MenuMCM.aspx	Companies
Manufacturing	Planning	MfgDemandPlanning.aspx	Demand planning
Manufacturing	Realization	MfgIssueMaterials.aspx	Issue materials - Selection
Manufacturing	Realization	MfgIssueReversalHistory.aspx	Issue reversal history
Manufacturing	Realization	MfgIssueReversals.aspx	Issue reversal - New
Manufacturing	Operations	MfgOperation.aspx	Operation
Manufacturing	Operations	MfgOperations.aspx	Overview: Operations
Manufacturing	Realization	MfgOperationsToPurchaseOrders.aspx	Overview: Purchase subcontracts
Manufacturing	Work centres	MfgProductionAreas.aspx	Overview: Production areas
Manufacturing	Planning	MfgShopFloorPlanning.aspx	Shop floor planning
Manufacturing	Planning	MfgShopOrder.aspx	Shop order
Manufacturing	Realization	MfgShopOrderComplete.aspx	Complete: Shop orders
Manufacturing	Realization	MfgShopOrderFinishItem.aspx	Finish shop orders
Manufacturing	Realization	MfgShopOrderFinishItemReturnHistory.aspx	Finish reversal history
Manufacturing	Realization	MfgShopOrderFinishItemReturns.aspx	Finish reversal - New
Manufacturing	Planning	MfgShopOrders.aspx	Overview: Shop orders
Manufacturing	Realization	MfgShopOrderStockReceiptPrint.aspx	Print shop order stock receipts
Manufacturing	Planning	MfgShopPapersPrint.aspx	Print shop paper
Manufacturing	Realization	MfgSubcontractDeliveryPrint.aspx	Print subcontract delivery notes
Manufacturing	Work centres	MfgWorkcenter.aspx	Work centre
Manufacturing	Work centres	MfgWorkcenters.aspx	Overview: Work centres
Person	Payroll	PrlCheckPeriods.aspx	Overview: Check

Person	Payroll	PrlCompGroupCLAS.aspx	Overview|Employment conditions groups
Person	Payroll	PrlComponentsDiv.aspx	Overview: Calculations
Person	Payroll	PrlControlRegister.aspx	Control Register
Person	Payroll	PrlCorrections.aspx	Overview: Correction requests
Person	Payroll	PrlEmplCostRegulation.aspx	Overview: Employment cost regulation
Person	Payroll	PrlEntries.aspx	Variable mutations
Person	Payroll	PrlHistoricalData.aspx	Overview: Historical entries
Person	Employees	prlMassUpdateSelection.aspx	Mass update : Selection
Person	Payroll	PrlPayments.aspx	Overview: payments
Person	Payroll	PrlPaySlips.aspx	Payroll slips
Person	Payroll	PrlProcessesOverview.aspx	Payroll process management
Person	Payroll	PrlRepJournalEntries.aspx	Journal entries
Person	Payroll	PrlReportsOverview.aspx	Overview reports
Person	Payroll	PrlTaxDeclarations.aspx	Overview: Tax declaration
Person	Payroll	PrlTaxReturns.aspx	Payroll taxes declaration
Person	Payroll	PrlVariableMutations.aspx	Variable mutations
Person	Payroll	PrlWagelist.aspx	Wage list
Person	Payroll	PrlYearEndOVerview.aspx	Year process management
Person	Payroll	PrlYearOverview.aspx	Income statements
Person	Payroll	PrlYearPEriods.aspx	Generate: Period-date table
Project	Projects	ProBucketPlanningBoard.aspx	Planning Board
Project	Entry	ProDayTransactions.aspx	Daily entry
Project	Invoices	ProInvoiceSpecification.aspx	Print invoice specification
Project	Projects	ProMyPlanning.aspx	My Planning
Project	Projects	ProProject.aspx	Project
Project	Projects	ProProjects.aspx	Overview: Projects
Project	Projects	ProRepBillability.aspx	Report: Billability
Project	Entry	ProTimeCostApproval.aspx	Approve
Project	Entry	ProWeekTransactions.aspx	Weekly entry
Sales	Subscriptions	PsnGenerateInvoices.aspx	Generate invoices
Sales	Subscriptions	PsnPrintSubscriptions.aspx	Print subscriptions
Sales	Subscriptions	PsnProlongation.aspx	Renew: Subscriptions
Sales	Subscriptions	PsnSubscription.aspx	Create: Subscription
Sales	Subscriptions	PsnSubscriptions.aspx	Overview: Subscriptions
Sales	Subscriptions	PsnSubscriptionSearch.aspx	Search: Subscriptions
Sales	Subscriptions	PsnUpdatePrices.aspx	Update subscription prices
Purchase	Purchase orders	PurBalanceList.aspx	Status overview: Purchase orders
Purchase	Purchase orders	PurEntryPurchOrder.aspx	Purchase order
Purchase	Items	PurPurchAnalysis.aspx	Purchase analysis
Purchase	Purchase invoices	PurPurchInvoices.aspx	Overview: Direct purchase invoices
Purchase	Purchase orders	PurPurchOrders.aspx	Overview: Purchase orders
Purchase	Goods receipts	PurPurchOrderSelect.aspx	Goods receipt - New
Purchase	Purchase orders	PurPurchOrdersPrint.aspx	Print purchase orders

Purchase	Goods Receipts	PurPurchOrdersToBeInvoiced.aspx	To be billed
Purchase	Returns	PurReceiptReturns.aspx	Print purchase return
Purchase	Goods receipts	PurReceipts.aspx	Print receipt slip
Purchase	Returns	PurReceiptSelect.aspx	Purchase return - New
CRM	Quotations	QuoEntryQuotation.aspx	Quotation
CRM	Quotations	QuoPrintQuotations.aspx	Print quotations
CRM	Quotations	QuoQuotations.aspx	Overview: Quotations
CRM	Quotations	QuoQuotationsApprove.aspx	Approve
CRM	Accounts	RepAccountCard.aspx	Transactions: Debtors
Financial	Costs units	RepAnalytical.aspx	Analytical values
Financial	Entries	RepEntryReport.aspx	Entry report
Financial	G/L Accounts	RepGLCardReport.aspx	General ledger transactions
Person	Payroll	RepPayrollPayments.aspx	Payment advice list
Person	Payroll	RepPrlAnnualReport.aspx	Year overview employee
Person	Payroll	RepPrlJrnlEntries.aspx	Journal entries
Person	Employees	RepPrlNoPrvtCar.aspx	Statement of no private use car
Person	Employees	RepPrlTaxStatement.aspx	Statement payroll taxes
Person	Payroll	RepPrlWagelist.aspx	Wage lists
CRM	Leads	SFALeadLifeTime.aspx	Lead Lifetime
CRM	Leads	SFALeads.aspx	History Lead
CRM	Opportunities	SFAOpportunities.aspx	Overview: Opportunities
CRM	Opportunities	SFAOpportunity.aspx	Opportunity
Project	Invoices	SlsInvoicePlanningAccounts.aspx	Proposals
Sales	Sales Orders	SlsMergeSalesInvoices.aspx	Merge invoices
Sales	Sales Orders	SlsPickingListPrintOnly.aspx	Pickinglist
Sales	Sales Orders	SlsPickingLists.aspx	PickingLists
Sales	Sales Orders	SlsRetToBeCredited.aspx	To be credited
Sales	Sales Orders	SlsSalesMargin.aspx	Sales margin (Summary)
Sales	Sales Orders	SlsSalesOrderEntry.aspx	Create: Sales order
Sales	Sales Orders	SlsSalesOrderPicklist.aspx	Sales order: Picking List
Sales	Sales Orders	SlsSalesOrders.aspx	Overview: Sales orders
Sales	Sales Orders	SlsSalesOrdersApprove.aspx	Approve
Purchase	Purchase Orders	SlsSalesOrderSelectToCreatePurchOrders.aspx	Back to back order
Sales	Sales Orders	SlsSalesOrdersPrint.aspx	Print sales orders
Sales	Sales Orders	SlsSalesordersTobeInvoicedAdv.aspx	To be invoiced
Sales	Sales Orders	SlsShipmentReturns.aspx	Sales return - New
Sales	Sales Orders	SlsShipmentReturnsPrint.aspx	Sales return history
Sales	Sales Orders	SlsShipments.aspx	Goods delivery - New
Sales	Sales Orders	SlsShipmentsPrint.aspx	Delivery history
Sales	Sales Orders	SlsStatusList.aspx	Status overview: Sales orders
Financial	Currency	SysCurrencies.aspx	Overview: Exchange rates
Person	Employees	WizNewEmployee.aspx	New: Employee

Appendix B. Deleted Aspx'es

Aspx	Reason
AssCDioxide.aspx	Country Specific
AssHistorySheet.aspx	Country Specific
ConMutations.aspx	Functionality for Resellers Only
CRMCreditChecker.aspx	Credit Check
CrmSocialBusinessTweets.aspx	Too General
FinBalanceList.aspx	Too General
FinBalanceReport.aspx	Country Specific
FinBWA.aspx	Country Specific
FinCentralizeGLAccount.aspx	Country Specific
FinDebtorStatements.aspx	Country Specific
FinEmpCostRegulations.aspx	Country Specific
FinEUSalesListBECorr.aspx	Country Specific
FinExpenseNonDed.aspx	Country Specific
FinPrivateUse.aspx	Country Specific
FinSupplierPaymentReporting.aspx	Report
FinTransactions.aspx	Too General
FinWithholdingModel111.aspx	Country Specific
FinWithholdingModel115.aspx	Country Specific
InvItemMRPOverview.aspx	Report
InvStockAgingAnalysis.aspx	Report
InvStockLocations.aspx	Report
InvWhereUsed.aspx	Report
IppSalesCreditMemos.aspx	Too specific for dashboard
IppSalesInvoices.aspx	Too specific for dashboard
LicHostingCustomers.aspx	Functionality for Resellers Only
LicResellerLinks.aspx	Functionality for Resellers Only
MenuPortal.aspx	Expresses no specific information
MfgBackflushOverview.aspx	Report
MfgIndentedBillOfMaterials.aspx	Report
MfgProductionAnalysis.aspx	Report
MfgShopOrderCost.aspx	Report
MfgShopOrderResults.aspx	Report
MfgShopOrderStatus.aspx	Report
PrlPayrollExport.aspx	Expresses no specific information
ProCustomerTrans.aspx	Report
ProRepCapacity.aspx	Report
ProRepCapacityForecast.aspx	Report
ProRepCosts.aspx	Report
ProRepPrepaidBalance.aspx	Report
ProRepScheduleAccountability.aspx	Report

ProRepWorkInProgress.aspx	Report
ProTimeCostTransactions.aspx	Report
PsnInvoiceForecastReport.aspx	Report
PsnReportBase.aspx	Report
SlsSalesAnalysis.aspx	Report
WizGraydonExport.aspx	Credit Check
XBRLInstanceDocuments.aspx	Country Specific

Appendix C. Case Study Results

Below all individual evaluation sessions are described. In the result tables the components are referred to by a number. The components belonging to these numbers are explained in the table below, the color codings show what components belong to the same sub category.

Component Number	Component Description
1	Outstanding items Receivable
2	Current Balance Bankaccounts
3	Contact Persons
4	Exchange Rates
5	Male/Female ratio
6	Birthdays
7	Number of invoice proposals
8	Written hours per project
9	Written hours per week
10	Billability
11	Revenue, costs, result
12	Total inventory value
13	Top items by revenue
14	Contractttypes
15	FTE Ratio
16	Employees per division
17	Number of proejcts per manager
18	Numer of projects this year and last year
19	Revenue per project this year and last year
20	Total amount in euros purchase invoices current year
21	Total amount of purchase invoices
22	Purchase invoices per GL Account
23	Purchase invoices per supplier
24	Returns
25	Sales orders: quantity
26	Sales orders: amount in euros
27	Number of active subscriptions
28	# of unprocessed entries

Wholesale user 1

Max number of components: 11

Percentage of correct suggestions (component): 81.82% (9/11)

Percentage of correct suggestions (subcat): 71.43% (5/7)

Component	Suggested by the model	Chosen by the subject
1	X	
2	X	X
3	X	X
4		
5		
6		X
7		
8		
9		
10		
11		X
12	X	
13	X	X
14		
15		
16		
17		
18		
19		
20	X	X
21	X	X
22	X	X
23	X	X
24		
25	X	X
26	X	X
27		
28		

	#	Reason
Extra Components:	6	Nice to know about birthdays
	11	Most important company figures
Did not pick:	1	Has no outstanding items, delivers after payment
	12	Inventory is not useful because there is no inventory. Buys when product is already sold.

Missing components: Urls to certain pages in the system.

Interesting quotes/findings: “Created own dashboard, my cockpit, your dashboard is similar but misses url’s to pages”

Wholesale user 2

Max number of components: 15

Percentage of correct suggestions (component): 80.00% (12/15)

Percentage of correct suggestions (subcat): 90.91% (10/11)

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3	X	X
4		
5		
6		
7	X	X
8		
9		
10		X
11	X	X
12	X	X
13	X	X
14		
15		
16		
17		
18		X
19		X
20	X	X
21	X	X
22	X	X
23	X	
24	X	X
25	X	
26	X	
27		
28	X	X

	#	Reason
Extra Components	10,18,19	Interesting to know about projects (note: User does not have project module)
Did not pick	23	only 1 supplier so per supplier is irrelevant,
	25,26	No outstanding sales orders because orders are handled immediately

Missing components: -

Interesting quotes/findings: Dashboard is more relevant to her than the standard dashboards

Wholesale user 3

Max number of components: 9

Percentage of correct suggestions (component): 66.67% (6/9)

Percentage of correct suggestions (subcat): 75.00% (6/8)

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3	X	
4		
5		
6		X
7	X	X
8		
9		
10		
11	X	X
12		
13	X	
14		
15		
16		
17		
18		
19		
20		
21		
22		X
23		
24		X
25	X	X
26	X	
27		
28	X	X

	#	Reason
Extra Components	22,24,6	Interested in this information
Did not pick	3	Contact persons are not stored in exact
	26	Only quantity of sales orders is important, not amount in euros
	13	Does nothing with items within exact, only changing their prices

Missing components: List of item prices

Interesting quotes/findings: -

Wholesale user 4

Max number of components: 8

Percentage of correct suggestions (component): 87.50% (7/8)

Percentage of correct suggestions (subcat): 85.71% (6/7)

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3	X	X
4		
5		
6		
7	X	X
8		
9		
10		
11	X	X
12		
13		
14		
15		
16		
17		
18		
19		X
20		
21		
22		
23		
24		
25	X	X
26	X	X
27		
28	X	

	#	Reason
Extra Components	19	Interesting to know about project revenues
Did not pick	28	Entries are not processed on request of the accountant

Missing components: -

Interesting quotes/findings:

-Use the standard dashboards a lot, but personalized like prototype would be appreciated

-Is a Professional services company but uses wholesale package

Wholesale user 5

Max number of components: 9

Percentage of correct suggestions (component): 88.89% (8/9)

Percentage of correct suggestions (subcat): 87.50% (7/8)

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3		X
4		
5		
6		
7	X	
8		
9		
10		
11	X	X
12	X	X
13	X	X
14		
15		
16		
17		
18		
19		
20		
21		
22		
23		
24		
25	X	X
26	X	X
27		
28	X	X

	#	Reason
Extra Components	3	Contact persons is interesting to see in dashboard
Did not pick	7	Does not want # of proposals but amount in euros

Missing components: invoice proposals amount in euros instead of # of proposals

Interesting quotes/findings: -

Manufacturing user 1

Max number of components: 10

Percentage of correct suggestions (on component): 80.00% (8/10)

Percentage of correct suggestions (on subcat): 83.33% (5/6)

Component	Suggested by the model	Chosen by the subject
1		
2		
3	X	X
4		
5		
6		X
7		
8		
9		
10		
11		X
12	X	X
13	X	
14		
15		
16		
17		
18		
19		
20	X	X
21	X	X
22	X	X
23	X	X
24	X	X
25	X	X
26	X	
27		
28		

	#	Reason
Extra Components:	6	Forgot colleague X's birthday today
	11	Important information to know about
Did not pick:	13	Only sell 2 products, so top 10 products is irrelevant
	26	Only number of outstanding sales orders is interesting, amount is predictable from this

Missing components: -

Interesting quotes/findings: -

Manufacturing user 2

Max number of components: 4

Percentage of correct suggestions (component): 25.00% (1/4)

Percentage of correct suggestions (subcat): 100.00% (1/1)

Component	Suggested by the model	Chosen by the subject
1		
2		
3		
4		
5		
6		
7		
8		
9		X
10		
11		X
12		
13		
14		
15		
16		
17		
18		
19		
20	X	
21	X	
22	X	
23	X	X
24		
25		X
26		
27		
28		

	#	Reason
Extra Components	25,11,9	Is not work/function related, just interested in this information about the company
Did not pick	20,21,22	Only wants to see purchase orders per supplier, is all the info he needs

Missing components: Returns per supplier, he is in quality assurance so interesting who the “good” and “bad suppliers are”
Interesting quotes/findings: -

Manufacturing user 3

Max number of components: 12

Percentage of correct suggestions (component): 10/12 = 83.33%

Percentage of correct suggestions (subcat): 8/10 = 80.00%

Component	Suggested by the model	Chosen by the subject
1	X	X
2		X
3		
4		
5		
6		
7	X	X
8		
9		
10		
11	X	X
12	X	X
13	X	X
14		
15		
16		
17		
18		
19		
20	X	X
21	X	
22	X	X
23	X	
24		X
25	X	X
26	X	X
27		
28	X	X

	#	Reason
Extra Components	2	Good to know current balance
	24	# of returns is interesting
Did not pick	21,23	Amount in euros and per gl account is sufficient

Missing components: Budgets

Interesting quotes: “The dashboard isn’t perfect but it’s a good start”

Manufacturing user 4

Max number of components: 15

Percentage of correct suggestions (component): 13/15 = 86.67%

Percentage of correct suggestions (subcat): 8/10 = 80.00%

Component	Suggested by the model	Chosen by the subject
1	X	X
2		
3	X	X
4		
5	X	
6	X	X
7		X
8		
9		
10		
11	X	X
12		X
13	X	X
14		
15		
16		
17	X	X
18	X	X
19	X	X
20	X	X
21	X	X
22	X	
23	X	X
24		
25	X	X
26	X	X
27		
28		

	#	Reason
Extra Components	7	Not relevant to this employee
	12	Not relevant to this employee
Did not pick	22	G/L account is not necessary if he can see purchase orders per supplier
	5	This is known because only 6 people work here

Missing components: -

Interesting quotes: -

Manufacturing user 5

Max number of components: 15

Percentage of correct suggestions (component): 12/15 = 80.00%

Percentage of correct suggestions (subcat): 8/9 = 88.89%

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3		
4		
5		
6		
7		X
8		
9		
10		X
11	X	X
12	X	X
13	X	X
14		
15		
16		
17	X	
18	X	
19	X	
20	X	X
21	X	X
22	X	X
23	X	X
24		X
25	X	X
26	X	X
27		
28	X	X

	#	Reason
Extra Components	7	
	10	
	24	
Did not pick	17,18,19	Does not work with projects

Missing components: -

Interesting quotes/findings : Users says they do not work with projects, but project related page views represent a great portion of this users activity log.

PSA User 1

Max number of components: 18

Percentage of correct suggestions (component): 14/18 = 77.78%

Percentage of correct suggestions (subcat): 10/12 = 83.33%

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3	X	X
4		
5	X	
6	X	
7	X	X
8	X	X
9	X	X
10	X	X
11	X	X
12		
13	X	
14		
15		
16		
17	X	
18	X	X
19	X	X
20		
21		
22		
23		
24		
25	X	X
26	X	X
27	X	X
28	X	X

	#	Reason
Extra Components	-	
Did not pick	5	Company is too small to worry about this
	6	Company is too small to not know this by head
	13	Only sells “Hours” so items are not that relevant
	17	Only 1 manager

Missing components: -

Interesting quotes: The suggested dashboard is similar to his ‘my cockpit’ dashboard

PSA User 2

Max number of components: 13

Percentage of correct suggestions (component): 10/13 = 76.92%

Percentage of correct suggestions (subcat): 8/8 = 100

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3	X	X
4		
5		
6		
7		
8	X	X
9	X	
10	X	
11	X	X
12		
13		
14		
15		
16		
17	X	
18	X	X
19	X	X
20		
21		
22		
23		
24		
25	X	X
26	X	X
27		
28	X	X

	#	Reason
Extra Components	-	
Did not pick	9	Hours are not monitored that strictly, only per project
	10	So billability is also not relevant
	17	There is only 1 manager

Missing components: -

Interesting quotes: -

PSA User 3

Max number of components: 13

Percentage of correct suggestions (component): 11/13 = 84.62

Percentage of correct suggestions (subcat): 7/8 = 87.50

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3		
4		
5		
6		
7		
8		X
9		
10		
11	X	X
12		
13		X
14		
15		
16		
17	X	
18	X	X
19	X	X
20	X	X
21	X	X
22	X	X
23	X	X
24		
25	X	
26	X	X
27		
28	X	X

	#	Reason
Extra Components	8	Hours are interesting, although company does not track hours in exact yet
	13	Top items would be interesting, but split on hours and products
Did not pick	17	Only 1 manager
	25	Only amount in euros is interesting

Missing components: -

Interesting quotes: Revenue split on hour sale, and item sale. Company sells both

PSA User 4

Max number of components: 14

Percentage of correct suggestions (component): 85.71% (12/14)

Percentage of correct suggestions (subcat): 88.89% (8/9)

9/10

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3		
4		
5		
6		
7	X	X
8	X	X
9	X	X
10	X	X
11	X	X
12		
13	X	X
14		
15		
16		
17	X	
18	X	X
19	X	X
20		X
21		X
22		
23		
24		
25	X	X
26	X	X
27		
28	X	

	#	Reason
Extra Components	20	
	21	
Did not pick	28	
	17	

Extra Picks: 20, 21

Didn't Pick: 28,17

Missing components: Budgets

Interesting quotes: My cockpit, has seen it, but not clear how it works

PSA User 5

Max number of components: 6

Percentage of correct suggestions (component): 83.33% (5/6)

Percentage of correct suggestions (subcat): 100.00% (2/2)

Component	Suggested by the model	Chosen by the subject
1		
2		
3		
4		
5		
6		
7		
8	X	X
9	X	X
10	X	X
11		X
12		
13		
14		
15		
16		
17	X	X
18	X	
19	X	X
20		
21		
22		
23		
24		
25		
26		
27		
28		

	#	Reason
Extra Components	11	Interesting information on company results
Did not pick	18	Have a select number of customers, that all have multiple projects a year, but those can be short or long, number of projects is not as relevant as revenue

Missing components: Billability per Person.

Interesting quotes: -

Master Thesis

The Automated Generation of User Specific Dashboards in a Multi-tenant ERP Product

Jasper van de Wiel

Master Business Informatics
Dept. of Information and Computing Science
Utrecht University

Utrecht University

Author	Jasper van de Wiel
	J.M.J.vandewiel@students.uu.nl
	Master Business Informatics
	Utrecht University
First Supervisor	dr. Slinger Jansen
	Slinger.jansen@uu.nl
	Department of Information and Computing Sciences
	Utrecht University
Second Supervisor	dr. Marco Spruit
	M.R.Spruit@uu.nl
	Department of Information and Computing Sciences
	Utrecht University
External Supervisor	Ruben Mijwaart
	Software Product Manager
	Exact Software Nederland B.V.
Eternal Supervisor	Jeroen Gruppen
	Software Engineering Manager
	Exact Software Nederland B.V.

I. Abstract

Users of ERP software are typically presented with a one size fits all product, even though each user has his own way of working. Customization of functionality by software producing organizations is a complex manual task. In an attempt to find a solution for this problem this research is focused on linking usage patterns to users of SaaS ERP systems, based on their ERP system usage data. These usage patterns are used in order to automatically provide the users with business intelligence dashboards, which are tailored to their specific information needs. In order to find a way to discover different usage patterns based on usage data the following research question has been formulated: “How can a model of user information needs be developed from ERP usage statistics in order to generate user specific BI dashboards?”

This research contributes to the body of scientific knowledge on the subject of personalized BI dashboards by providing a method. This method can be used in order to develop a model that provides a systematic way to discover different usage patterns based on the usage data of an ERP system. This model has been created by performing web usage mining on a big dataset containing over 15.5 million user actions in a SaaS ERP product. In order to find a method for creating a model of usage patterns a number of techniques including process mining and association rule mining have been studied. The final method used to perform web usage mining is content enhanced clustering. The clustering technique used is Formal Concept Analysis. The data consists of usage data, semantic data and usage data of the ERP system.

The model has been evaluated with end-users of a large SaaS ERP software product by using a prototype of a dashboard generator making use of the developed model in an extensive case study. The results of the case study show that the generated dashboards are useful, accurate, and save the end user time in composing their own. The research process and the findings of this research are documented in this document.

II. Acknowledgements

This document is the result of hours of hard work, consisting of research on the application of usage data in order to provide a personalized experience to users of ERP systems. The research process that is the basis for this thesis has been a great educational experience for me. I have performed literature research on a variety of interesting topics, I have had the chance to experiment with multiple datamining techniques, I have spoken to many experts in the field of ERP systems, developed a prototype, and evaluated this prototype with end-users of an ERP system.

I would like to express my gratitude to my first supervisor dr. Slinger Jansen for his motivating and constructive criticism during our meetings and digital conversations. I would like to thank dr. Marco Spruit for his feedback on this thesis, and his sharp eye. Thanks also to my external supervisor Ruben Mijwaart, former product manager at Exact, for his guidance on the direction of my research. Of course, a big thank you to all other people at Exact that made it possible for me to perform my research at an innovative software company. A special thanks to my colleagues of the connectivity team that I have had the pleasure of working with for the past year.

Last but not least, I want to thank my friends and family, who have all supported me during my research. A special thanks to Dieuwertje, for supporting me morally, and more importantly for bearing with me through this time-consuming period.

Have a good read!

Jasper van de Wiel
Tilburg, January 2017.

III. Table of contents

I. Abstract ..... **V**

II. Acknowledgements ..... **VII**

III. Table of contents ..... **IX**

1. Introduction **1**

1.1 Background 1

1.2 Research Focus 1

1.3 Overall Research Aim and Individual Research Objectives 2

2. Research Method **4**

2.1 Research Overview 4

2.2 Design Science 6

2.3 Literature Review Protocol 8

2.4 Model Development 8

2.5 Evaluation 9

3. Theoretical Background **11**

3.1 Software as a Service 11

3.2 ERP as a Service 13

3.3 Multi-Tenancy 13

3.4 Business Intelligence 14

3.5 Customization & Configuration 15

3.6 Web Usage Mining 16

4. Model Development - Data Gathering and Pre-processing **23**

4.1 Data Gathering 23

4.2 Data Pre-Processing 25

5. Model Development - Methods Considered for Pattern Discovery **28**

5.1 Association Rule Mining 28

5.2 Process mining 29

5.3 Formal Concept Analysis 30

5.4 Chapter Conclusion 33

6. Model Development – Pattern Discovery **34**

6.1 The Dataset 34

6.2 Performing Content Enhanced FCA 35

6.3 Chapter Conclusion 40

7. Model Development – Pattern Analysis 41

7.1 Pattern Analysis 41

7.2 Expanding the model 43

7.3 Chapter Conclusion 45

8. Prototype 46

9. Evaluation 50

9.1 Case Study Protocol 50

9.2 Case Study Participants 50

10. Results 51

10.1 Case Study Results per Package 51

10.2 Chapter Conclusion 53

11. Discussion & Future Work 54

11.1 Discussion 54

11.2 Future work 55

12. Conclusion 56

References 59

Appendices 63

Appendix A. Aspx'es, Related (Sub)Menu Items, and Title Elements 63

Appendix B. Deleted Aspx'es 68

Appendix C. Case Study Results 69

1. Introduction

1.1 Background

More than ten years ago, Kaplan (2005) observed that a great number of software companies were trying to make a switch from on premise deployment towards the Software as a Service business model. The SaaS model is especially interesting for Small and Medium businesses (SMEs) because these enterprises usually have a small IT budget (Bezemer, Zaidman, Platzbeecker, Hurkmans, Hart, 2010; Wang, Guo, Gao, Sun, Zhang, An, 2008). Gartner uses the number of employees and a business's revenue as a measure to classify a business as an SME. Small businesses are businesses with less than 100 employees, and midsize companies have a max of 999 employees. For revenue, the boundaries lay with respectively $50 million and $1 billion in annual revenue. SMEs have a strong need for business software, for instance Enterprise Resource Planning (ERP) systems. In particular business software as a service is interesting to SMEs because the on premise deployment of for instance an ERP system comes at a high price (Makkar & Bist, 2012).

A popular architectural pattern used by ERP providers for delivering software as a service is multi-tenancy. With the use of the multi-tenancy model, only one instance of the software is running on the side of the software provider. All the customers (tenants) of the software provider access the same instance of the software (Bezemer & Zaidman, 2010). A tenant usually does not refer to one person, but rather to (part of) an organization. Each tenant has its own specific requirements to the software (Kabbedijk & Jansen, 2011). Multi-tenancy enables the software provider to benefit fully from the economy of scale by making serving multiple clients more scalable and cost-effective.

One key feature of ERP systems is providing its users with reports on business information. This is realized by implementing some form of Business intelligence (Later referred to as BI) functionality. In ERP systems, BI is often put into practice in the form of dashboards. Dashboards are data visualization tools that provide visual displays of important information, preferably on a single screen so that the information can be processed and explored quickly (Palpanas, Chowdhary, Mihaila & Pinel, 2007). A dashboard can for instance help to visualize a trend in the data that is hard to see from the data itself by showing the data in a graph. Dashboards play a big role in today's business environments. According to Tutunea and Rus (2012), dashboards have gained ground in recent years due to the lowering costs of implementing BI.

1.2 Research Focus

Despite the large adaptation of dashboards in businesses the design and implementation of dashboards is still prone to failure (Aldridge, 2008; Few, 2006). A big cause of the failure of dashboards is the dashboard not showing the right information (Few, 2006).

Aligning dashboards with a company's business goals and characteristics is challenging

(Aldridge, 2008; Eckerson, 2003; Kawamoto & Mathers, 2007; Watson et al., 2006, 2007).

To help a user grasp what is on the dashboard the information should be linked to relevant concepts in the business context (Frank, Heise & Kattenstroth, 2009). This misalignment is often caused by providing a dashboard that is not tailored to the specific business characteristics of a company. If the BI solution does not match with the user's characteristics and goals, it will just be a cool gimmick instead of a useful tool providing its users with relevant information about their companies' performance. This in turn leads to users looking for their information in other places (Eckerson, 2010).

In accordance with dashboards, ERP systems also need to be tailored to the needs of its users (Kumar & Hillegersberg, 2000). On top of that, multi-tenant systems should also be tailored to the needs of its users. Multi-tenant systems need a high degree of tailorization (Bezemer et al., 2010b). The need for tailoring in multi-tenant systems follows from the difference between the multi-user architecture and the multi-tenant architecture (explained in section 1.1). The single software instance hosted by the software provider should be tailored to the requirements of the different tenants using this single instance (Kwok, Nguyen, and Lam, 2008).

It is evident that for ERP systems, multi-tenant software, and dashboards it is important that the content provided to the user is tailored to the needs of the specific user. This means that a high degree of tailorization is certainly needed for tenant-specific reporting in multi-tenant ERP systems. Tailoring is a way of providing flexibility to the user. A problem with providing users with more flexibility in the software is that the usability of the system will decrease. This is the so-called flexibility-usability tradeoff (Steinebach, Guhathakurta & Hagen, 2009). The reason for the decrease in usability is the increasing number of options a user is confronted with in the user interface of a highly configurable software product. The background information in the previous and current section leads to the following problem statement:

SaaS ERP vendors do not have valid approaches to customize BI dashboards to the needs of individual users.

1.3 Overall Research Aim and Individual Research Objectives

If the process of customizing to the specific needs of a user can be automated in such a way that the user is provided with user specific reports, there is no need for flexibility on the user side. This means the usability problem is not relevant anymore. On top of that, the chance of failure of the reporting tool is also decreased because users will always get relevant reports. The overall aim of this research derived from above statements is: To find a way to automatically generate dashboards tailored to the information needs of individual employees of SMEs in a multi-tenant ERP software product. This overall research aim has been transformed into the following main research question:

RQ: How can a model of user information needs be developed from ERP usage statistics in order to generate user specific BI dashboards?

In order to structure the research, the main research question has been split into four sub research questions. The combination of answers to these sub questions answers the main research question.

SRQ1: How can the different usage patterns of SME employees while using ERP systems be identified?

This question is the first question that needs to be answered. In order to answer this question a number of different data mining methods that might be useful in order to identify different usage patterns have been studied. The next goal is to identify the different usage patterns of SME employees while using ERP systems. This leads to the second research question:

SRQ2: What are the different usage patterns of SME employees while using ERP systems?

If different users of ERP systems have different usage patterns, the next goal is to find out what the specific information needs of the different users expressed by the usage patterns are. This is the basis for the third research question.

SRQ3: How do the usage patterns express different information needs of SME employees while using ERP systems?

The formulation of this third research question suggests there is certainty that the information needs of employees of SMEs using ERP systems differ from each other. However, the answer to this research question might as well be they do not differ. This means there is a need to find out if employees of SMEs making use of ERP systems have different information needs, and if so, how they are different from each other. If the different users all have the same information needs, there is no need to generate dashboards configured to the specific information needs of each tenant. Because if every user has the same information needs a general dashboard is sufficient in covering the needs of all the users.

SRQ4: To what extent does a dashboard, generated based on usage patterns, satisfy the information needs of individual SME employees?

If the usage patterns and related information needs of ERP software users can be identified, a dashboard generation tool that provides each user with a dashboard, automatically generated based on the information needs of that user can be developed. When this tool has been developed, the dashboards generated by this tool can be evaluated in order to see if the information needs as identified by the generator are truly the information needs the user has.

A short overview of what research methods are used and how they are linked to the different research questions, is depicted in the research model in figure 1. The research methods are explained in more detail in the next chapter.

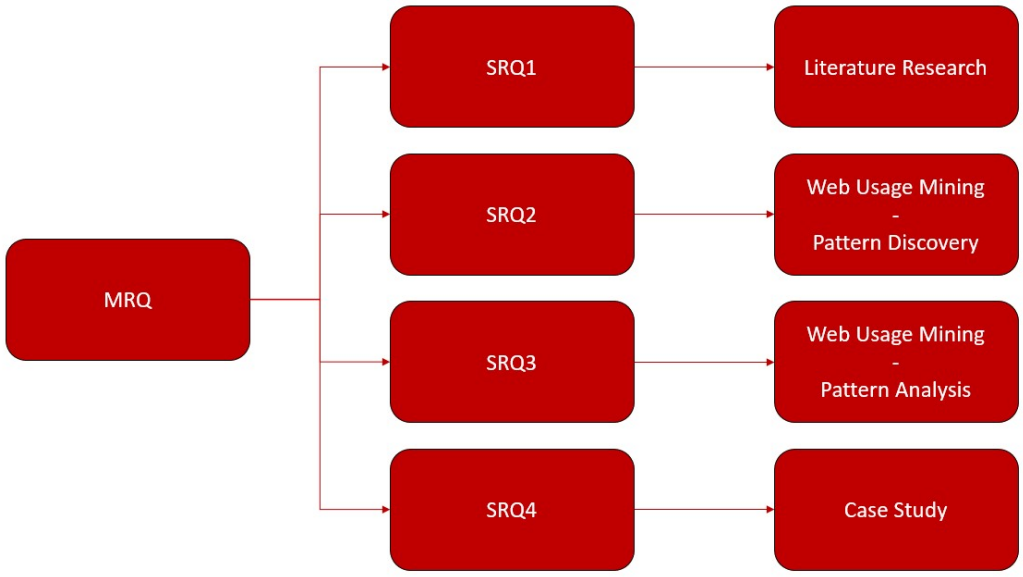

Figure 1. Research model

2. Research Method

2.1 Research Overview

In order to answer the research questions stated in section 1.3 the research has been conducted in four consecutive steps as depicted in the Process-Deliverable Diagram (PDD) in figure 2. This PDD shows a high-level overview of the research. A number of different methods are used throughout this research in order to answer the research questions in a scientifically sound manner. However, all steps are conducted by using an overarching research paradigm called Design Science. The design science paradigm has been employed using the framework as described by Peffers Tuunanen, Rothenberger and Chatterjee (2007), making use of the guidelines as proposed by Hevner, March, Park and Ram (2004), and taking into account the refinements to these guidelines as described by Wieringa (2009). The design science paradigm is suitable for this research because an organizational problem has been identified that needs to be solved. The need for the automatic generation of business intelligence dashboards tailored to the information needs of individual tenants making use of ERP software products has been identified. This need has been identified from scientific literature by performing a literature review, and is supported by a business need. An artefact has been developed, that contributes to solving the organizational problem. In addition to this, the research helps answering the main research question and thus, adds valuable findings to the current body of knowledge. These two reasons match exactly the criteria for using the Design Science paradigm.

Two different methods are used in order to gather the data for this research. One method is the case study research method. The other method is a data mining approach called web usage mining. The case study research method is used at different steps in the research and is applied by using a number of different techniques. For example, client interviews and expert interviews have been used in order to evaluate the model that follows from applying the data mining techniques, but also prototype development and thinking aloud sessions have been used in order to evaluate the deliverables created in this research. The company that acts as the case study company in this research provides the data used in the data mining tasks.

The design science paradigm, and the way in which it has been applied in this research, is explained in more detail in section 2.2. In section 2.3, the literature review protocol is presented. Section 2.4 elaborates on the ontology creation and section 2.5 is concerned with the evaluation techniques used during this research.

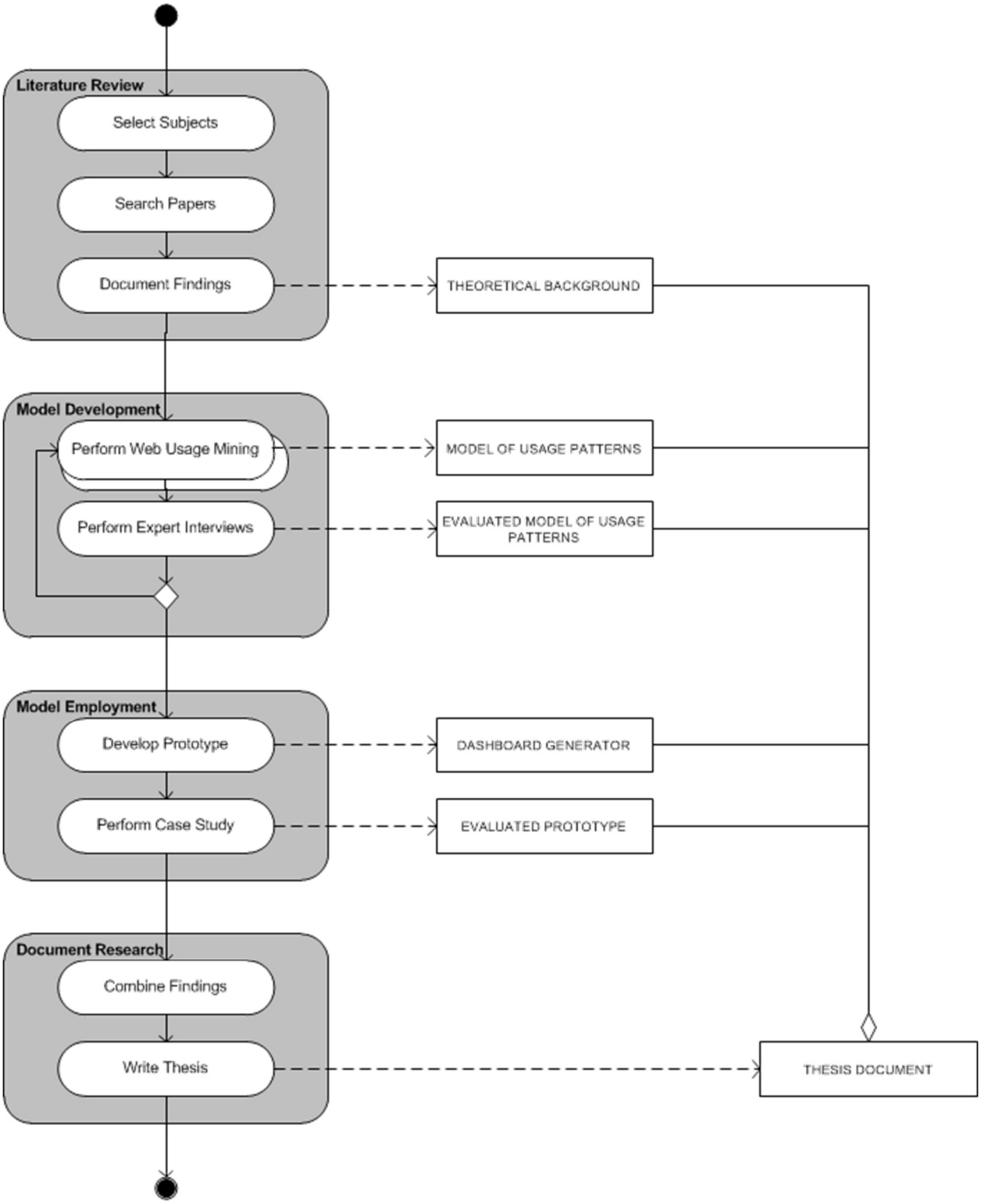

Figure 2. Research overview

2.2 Design Science

In order to execute the research, and answer the research questions, the design science paradigm has been applied. There are a number of ways of applying the design science paradigm, this research has been conducted using the design science research methodology (DSRM) developed by Peffers et al. (2007). This methodology is based on the design science research process model (DSRP) as defined by Peffers, Tuunanen, Gengler, Rossi, Huis, Virtanen and Bragge (2006). In addition, in order to conduct solid design science research the seven guidelines described by Hevner et al. (2004) are adhered to, taking into account the proposed refinements by Wieringa (2009).

In order to give shape to the design science research process the DS process model has been used, which is part of the DSRM as proposed by Peffers et al. (2007). The authors have developed a process for design science research for information systems that is consistent with earlier keystone literature on DS research. The process provides a nominal process model for performing design science research, and it provides a mental model for presenting and appreciating design science research for information systems. The consistency with earlier literature is accomplished by looking at leading literature on design science research. Seven representative DS papers (Archer, 1984; Eekels & Roozenburg, 1991; Hevner et al., 2004; Nunamaker, Chen, Purdin, 1991; Rosi & Sein, 2003; Takeda, Veerkamp, Tomiyama & Yoshikawam, 1990; Walls, Widmeyer & Sawy, 1992) have been compared and Peffers et al. conclude that the authors agree substantially on common elements. This results in the DS process model consisting of six steps in a nominal sequence. The model has successfully been applied by the authors themselves for at least four times, and has been applied by other researchers numerous times. The application of the method is considered successful if the paper describing the research has been published. The fact that the model has been used in successful research proves that the process model is nominal and that it provides a mental model for presenting design science research for information systems.

The process consists of the following six steps:

1. Problem identification and motivation;
2. Objectives of a solution;
3. Design and development;
4. Demonstration;
5. Evaluation;
6. Communication.

An adaptation of the original visualization of the DS process model by Peffers et al., applied to our research approach, is depicted in figure 3. In this research the problem centered approach has been followed by starting at the “Identify Problem & Motivate” step in the model. The problem has been identified and motivated by looking into scientific literature, this has been done briefly in the introduction of this thesis and in more detail in the literature review. The objectives of the solution are partly specified based on literature; the other part is specified based on expert interviews. The design and development step is brought into practice by developing a model making use of data mining. This model is demonstrated by putting it into practice in a prototype, which provides dashboards based on the model. This prototype has been put to practice in a case study. The model is evaluated by comparing the objectives of the prototype to the actual observed results from the use of the prototype in the case study. Finally, the communication step is completed by writing down our findings in this document.

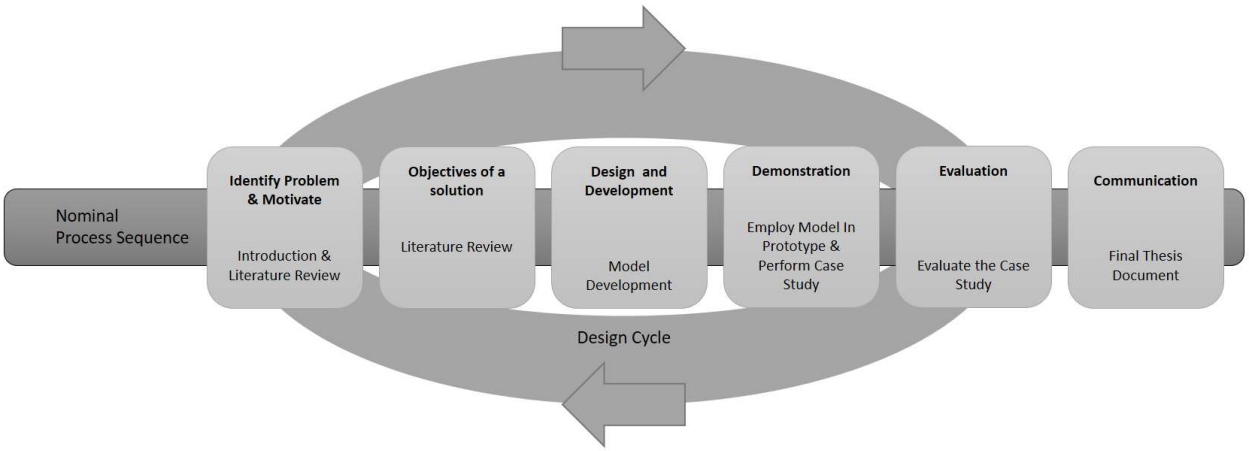

Figure 3. Adaptation of the DS process model

In addition to the process sequence by Peffers et al. (2007) the seven design science guidelines as proposed by Hevner et al. (2004) are adhered to, taking into account the refinements proposed by Wieringa (2009). In table 1 below, the design science approach of this research has been mapped out to the steps of the DS process model. In the right most column references to the guidelines of Hevner et al. (2004) have been placed at the steps these guidelines are applied to.

Table 1. DS process model mapped to activities and guidelines.

DS process model step	**Activities**	**Guideline (Hevner et al., 2004)**
1. Problem identification & motivation	This research started by identifying and **defining a problem statement** in chapter 1. The importance and need of this research has been motivated in chapter 1 as well. This has been done to a larger extent in the **literature review** documented in chapter 3.	Problem Relevance
2. Objectives of a solution	The objectives of the solution are also gathered from **literature**.	Problem Relevance
3. Design and development	The design and development of the artifact (**The Model**) is described in chapter 4, 5, 6 and 7.	Design as an artifact, Research Rigor, Design as a Search Process
4. Demonstration	The demonstration has been completed by bringing the model into practice in a **prototype** which provides users with user specific dashboards created based on the model. The prototype is described in chapter 8.	Design Evaluation, Design as a search process
5. Evaluation	The evaluation is performed in the form of a **case study** in which users of the SaaS ERP system use the developed artefact. This case study is described in chapter 9, and the results are described in chapter 10.	Design Evaluation, Research Rigor, Design as a search Process
6. Communication	The final **thesis document** is the way in which this research is communicated.	Research contribution, Communication of research

2.3 Literature Review Protocol

The first step of this research has been completed by performing a literature review. The literature review provides the necessary background information in order to answer the research questions in the next steps of the research. The fields of study of this research are 'Business Intelligence', 'Dashboards', 'ERP as a Service', 'Formal concept analysis', 'Multi Tenancy', 'Software as a Service', 'Tailoring', and 'Web usage mining'. A quick overview of how these subjects are related to the research proposed in this document has been provided in sections 1.1 and 1.2. In the literature review, each topic is discussed in more detail. A topic that is mentioned in the introduction, but is part of this literature review, is the topic of web usage mining.

In the literature review google scholar and the online computer science bibliography DBLP are used in order to gather relevant papers on the subjects of research. The first set of papers has been gathered based on (equivalents and/or synonyms of) the following search terms and combinations thereof: 'Software as a Service', 'Multi-tenant software', 'Business intelligence', 'Dashboards', 'Tailoring' and 'Web Usage Mining'. Some examples of what is meant with synonyms/equivalents are: 'SaaS' instead of 'Software as a Service' and 'BI' instead of 'Business Intelligence'. But also: 'Configured' instead of 'Tailored' and 'Reporting' instead of 'Dashboards'. Some examples of what is meant with "combinations thereof" are: 'Tailored dashboards', 'Configurable Business Intelligence' and 'BI in Multi-tenant software'.

More papers have been gathered by applying the forward and backward reference search method on the first set of papers. The backward reference searching method involves looking into the references that are cited in the first set of papers, in order to find additional interesting papers. The forward reference searching method is concerned with looking for papers that have cited one of the papers that are selected in the first set of papers. This means looking into publications created after the initial article has been created. After completing the search for papers based on the forward and backward reference search method a first selection of papers is made based on the titles of the papers. This lead to a smaller set of papers out of which a relevant subset has been selected by reading the abstracts. The last selection is based on the full text of the papers, after the last selection round, a set of papers remains that consists of papers that are relevant to the scope of this project. This set of papers has been used to write a thorough literature review on which the rest of this research project is based.

2.4 Model Development

In order to discover the different usage patterns of the users, web usage mining has been used. The first step performed in order to apply web usage mining is data gathering. The data has been gathered by querying the activity log of a big SaaS ERP product, and by scraping data from the pages of this ERP system. Next pattern discovery has been performed by applying formal concept analysis on the gathered data. Because usage data is combined with content and structure data, this method is called content-enhanced formal context analysis. The application of content-enhanced formal context analysis and pattern analysis results in different usage patterns based on the ERP usage of different users.

2.5 Evaluation

In the evaluation of the model of different users and their information needs, the first step is to perform expert interviews with experts in the domain of ERP systems. After the model has been evaluated with experts, a prototype of a dashboard generator has been developed. This prototype has been used in a case study in order to evaluate that the model is not only correct but that it can also be applied in order to generate meaningful dashboards for users of an ERP system. This has been validated by checking if the dashboards generated based on the model deliver valuable and relevant information to the users within the different usage patterns.

2.5.1 Expert Interviews

The initial model is based on findings in the data. However, in order to evaluate the model with expert knowledge, experts in the field have been interviewed. In order to do this, expert interviews have been conducted with software product marketeers, product managers, employees of a customer intelligence team and software developers. The experts all have experience working at a big SaaS ERP provider.

2.5.2 Prototype

The prototype has been developed using PHP. The prototype has been developed as an app for the app center of a big SaaS ERP product. The application pulls the app users' data from the SaaS ERP product via a REST API. This data is used in order to compare the users' usage data to the model, and link the user to a usage pattern in the model. The usage pattern the user is linked to prescribes what components should be shown in the dashboard.

The dashboard is generated by pulling the prescribed data from the user data by, again, making use of the REST API. And pushing the data into Microsoft power BI. The dashboard that is formed by doing this is shown to the user in the web app. Microsoft PowerBI has been used because it provides a quick and easy way to create an appealing and clear dashboard, and therefore is a suitable tool for prototyping.

2.5.3 Case study

The case study protocol is constructed by following the guidelines provided by Yin (2009). According to Yin (2009) there are three types of case studies: descriptive, exploratory, and explanatory case studies. Moreover, there are four kinds of case study designs: Single-case holistic designs, single-case embedded designs, multiple-case holistic designs, and multiple-case embedded designs. These four kinds of case study designs are depicted in figure 4 below. The distinction between the single- and multiple-case design depends, as the name suggests, on if a single case or multiple cases are being studied whilst conducting the case study.

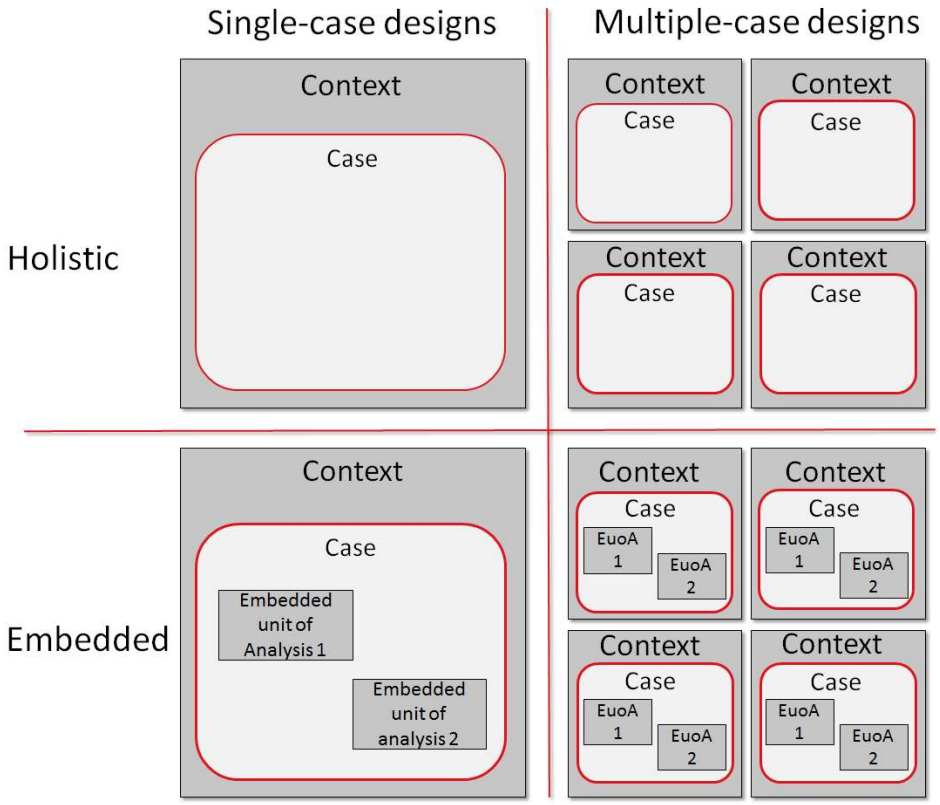

Figure 4. Case study designs (Yin, 2009)

The case study in this research is a single case embedded case study. A single case with multiple units of analysis has been studied. The case study is both exploratory and explanatory. The case study has been executed by evaluating the created prototype with a number of customers of a SaaS ERP product. The prototype has been evaluated by first providing the case study subjects with all possible dashboard components showing dummy data. The users are told to pick the components they would like to see in their personal dashboard. Next, a dashboard is shown that, according to the developed model, fits the information needs of this specific user. This dashboard contains the real user data. The components in this dashboard have been compared to the ones picked by the user in the first test. If there are any differences the user is asked if they still support their initial opinion, or if they like the generated dashboard better. The differences have been discussed thoroughly in order to check if the generated dashboard satisfies the user's needs.

3. Theoretical Background

In this chapter, the major topics involved in this research are discussed. The information for this chapter has been collected by executing the literature review as described in section 2.3. The topics covered in this literature review are software as a service, ERP as a Service, Multi-Tenancy, Business Intelligence, tailoring and web usage mining. The literature available on each of these topics has been reviewed in a separate section.

3.1 Software as a Service

SaaS is software that, contrary to on premise software, is accessed by the user via the internet (Dubey and Wagle, 2007). The on premise business model involves the deployment and use of software at the physical location of the customer. Whereas with the SaaS business model the software is hosted by the software vendor and the user accesses the software via the internet. So instead of buying a packaged software product the software is leased by the customer. The customer pays the software provider by for example a 'per-user' or 'per-month' subscription fee (Kaplan, 2007). As stated in the introduction, for the past ten years numerous software companies have been making a switch from on premise software to the SaaS business model. This switch has especially been focused on the enterprise software domain (Kabbedijk, Bezemer, Jansen, Zaidman, 2014). This switch became possible because of the advancing of technology and the rise of the web as the main medium for communication (Nitu, 2009). Customers were getting more used to the web and therefore became more comfortable doing business via the internet (Mertz, Eschinger, Eid & Pring, 2007).

Software vendors identified a possible gold mine by providing SMEs with a small IT budget with software via the SaaS model. SMEs often could not afford expensive on-premises or ASP software (Bezemer et al., 2010a; Wang et al., 2008). On premise software requires a big one-time investment at the moment of purchase and is expensive to maintain (D'souza et al., 2012). And ASP software was costly to maintain and set-up for software vendors, these costs were piped to the customers of ASP software (Nitu, 2009). SaaS applications do not require a big investment since the consumer usually pays a monthly fee (Kaplan, 2007), this is why SaaS is gaining this much popularity on the consumer side. Moreover, SaaS applications are cheaper to maintain for the provider, since there is only a single code base to maintain. This means lower costs for the provider and therefore lower costs for the customer. Another aspect which saves the customer money is the fact that SaaS applications are centrally managed at the side of the software provider, this means that the customer does not need to hire in-house experts on the software (Kaplan, 2007; Nitu, 2009; Wang, Laszewski, Younge, He, 2010). For software providers the decreased time to market is one of the main reasons why SaaS is gaining so much popularity (Aulbach, Grust, Jacobs, Kemper & Rittinger, 2008).

Where the benefits of SaaS are to a large extent agreed upon by the scientific community, there is no consensus on the definition of SaaS yet. Numerous definitions for SaaS have been provided in literature. Mäkilä, Järvi, Rönkkö and Nissilä (2010) capture eight of these definitions, and have identified "five distinct characteristics that are typically associated with SaaS". These five distinct characteristics are the following:

1. Product is used through a web browser;
2. Product is not tailor made for each customer;
3. The product does not include software that needs to be installed at the customer's location;
4. The product does not require special integration and installation work;
5. The pricing of the product is based on actual usage of the software (Mäkilä et al., 2010).

These characteristics are important characteristics of SaaS; however, the authors do not provide a unified definition for SaaS. Moreover, in our opinion the authors in their selection of papers defining SaaS missed a few important ones. The most important paper they missed is the paper by Chong and Carraro (2006). They give the following definition of SaaS: "Software deployed as a hosted service and accessed over the Internet."

As mentioned by the authors themselves this definition is a little simplistic, since it also describes simple, non-mature, SaaS solutions. However, the definition is in accordance with other authors' views on SaaS. For example, Buxmann, Hess and Lehmann (2008) describe SaaS as providing a software solution to customers as a service via the internet, in which the provider of the software hosts the software. And Dubey and Wagle (2007) write about delivering services that can be accessed and used via the internet. Thus, what is missing in the simple definition of Chong et al. is the notion that the provider and the user are different parties. Chong et al. complement their definition by providing a maturity model for SaaS products (Figure 5). This model provides four levels of SaaS maturity.

In the first maturity level as depicted in the top left corner of figure 5 every tenant has its own unique product provided by and hosted at the service provider. This means configuration is provided by different versions of the code for each tenant. This level of maturity is similar to the Application Service Provider (ASP) model. SaaS can actually be seen as the successor of ASP (Buxmann et al., 2008; Kaplan, 2007). The ASP software model was introduced in order to serve the same purpose as SaaS does now; it also made use of the principle that the software provider hosts the software that the customer can use by connecting to it via the internet (Papazoglou, 2003). However, as explained above the ASP model was more expensive than the SaaS model and therefore quickly improved to evolve to the SaaS model.

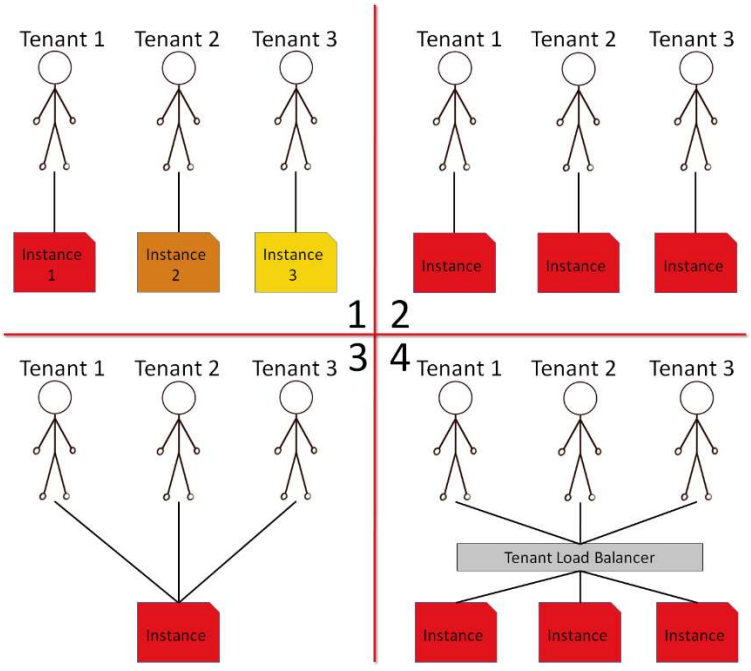

Figure 5. Software as a Service maturity model adapted from Chong & Carraro (2006).

In the next maturity level, depicted in the top right of figure 5, each tenant is provided with its own instance of the same code. Configuration is provided by offering configuration options so each tenant can customize its instance to their own preferences.

Where maturity level one and two can be seen as single-tenant architectures, maturity level three and four are using the so-called multi-tenant architecture. In the third maturity level, tenants make use of a single instance of the software product. Configurability is provided by making use of meta-data. The data of tenants is kept privately to each tenant. In order to make the multi-tenant architecture scalable a load-balanced farm of instances is hosted by the software provider, as can be seen in maturity level four.

In order to exploit the economy of scale the provider of SaaS should make sure their product adheres to the Multi-Tenancy architecture model (Mietzner, Metzger, Leyumann, Pohl, 2009; Bezemer et al., 2010b; Kwok et al., 2008; Kabbedijk et al., 2014). The multi-tenancy model is explained in more detail in section 3.3.

3.2 ERP as a Service

ERP Software fuses information throughout an entire enterprise to allow for a seamless flow of information (Engebrethson, 2012). This means ERP software provides companies with a way to manage their resources in an efficient and effective way (Fui-Hoon Nah, Lee-Shang Lau & Kuang, 2001). Resources can be material resources but also human resources and financial resources. ERP systems enable companies in doing so "by providing a total, integrated solution for the organization's information-processing needs." (Fui-Hoon nah et al., 2001). Since the year 2002 ERP vendors are recognizing and responding to the needs of companies to add BI functionality to their ERP systems (Griffin, 2007). This BI functionality is usually not as sophisticated as in standalone BI systems. (Chou, Tripuramalla & Chou, 2005) However, it suffices in providing smaller companies in their business intelligence needs. In section 3.4 BI is discussed in a more elaborate way, in this chapter more information is provided to motivate what is classified as "not as sophisticated" BI exactly.

Traditionally ERP software vendors have focused mainly on the large company segment of the market (Everdingen, Hillegersberg & Waarts, 2000). This focus has definitely borne fruit, since ERP systems are adopted by large enterprises up to a level of market saturation (Bernroider, Koch, 2001). Because of this market saturation ERP vendors are now focusing on SMEs (Haddara & Zach, 2011). SMEs feel the need for using ERP systems (Xu, 2011). However, as mentioned in previous sections of this thesis SMEs usually do not have a big IT budget and can therefore not afford the initial purchase of classic on-premises ERP software. However, if vendors offer an ERP that can be obtained by paying for example a monthly subscription fee, SMEs are able to obtain an ERP system even with their small IT budget. The way in which ERP vendors can do this is by offering a SaaS ERP solution (Makkar & Bist, 2012). To keep providing the software interesting for the vendor the vendor should be able to exploit the economies of scale. As discussed in section 3.1 this can be realized by employing the multi-tenancy architecture, multi-tenancy is elaborated upon in the next section of this chapter.

3.3 Multi-Tenancy

In chapter 1 of this thesis multi-tenancy was already introduced briefly. However, in order to gain a clear understanding of what multi-tenancy is exactly existing literature on the topic has been reviewed in order to find a good definition. The most recent definition found in scientific literature is by Kabbedijk et al. (2014) who define multi-tenancy as:

"Multi-tenancy is a property of a system where multiple customers, so-called tenants, transparently share the system's resources, such as services, applications, databases, or hardware, with the aim of lowering costs, while still being able to exclusively configure the system to the needs of the tenant."

Although this definition has been constructed by performing an extensive literature analysis we do not fully agree on the definition. Kabbedijk et al. (2014) argue that transparency should be included in the definition because it would otherwise be too generally applicable. According to the authors, there would be no difference between the multi-user model and the multi-tenant model if transparency is not included in the definition. In our opinion, the configurability part of the definition is already distinguishing the multi-tenant model from the multi user model. The difference between the multi-tenant model and the multi-user model is explained by Bezemer et al. (2010a) as follows: In the multi-user model the users are also using a shared application and database instance, however they have little configurability options. In the Multi-tenant model the tenants are using a shared application and database instance and have the possibility to configure the application, even to an extend in which the workflow of the application may differ per tenant. Another definition, albeit for a Multi-tenant application instead of just multi-tenancy, is provided by Bezemer et al. (2010a):

"A multi-tenant application lets customers (tenants) share the same hardware resources, by offering them one shared application and database instance, while allowing them to configure the application to fit their needs as if it runs on a dedicated environment."

This definition is almost identical to the one by Kabbedijk et al. but in our opinion, it's more accurate because it leaves the transparently part out of the definition. Bezemer et al. also provide a definition for what a tenant is exactly. They define a tenant as:

"The organizational entity which rents a multi-tenant SaaS solution. Typically, a tenant groups a number of users, which are the stakeholders in the organization."

Multi-tenancy's main benefit for the service provider is cost reduction, because the service provider only has to deploy one instance of the software and their hardware can be used more efficiently because not every tenant needs its own set of hardware (i.e. Servers). In addition, the utilization rate of the hardware is increased drastically when multiple tenants share the same IT infrastructure (Warfield, 2007; Kwok et al., 2008). This in turn provides a benefit for the customer, especially for SMEs since they do not need the computational power of a dedicated server (Bezemer, 2014).

3.4 Business Intelligence

As explained in the introductory chapter and in section 3.2 of this thesis some multi-tenant software systems offer their tenants with BI functionality. Before going into detail about the use of BI in ERP software systems, BI is defined. BI is a broad term that has a different definition to different people (Jones, 2009). The meaning of this term depends on the background of the one defining it (Sharda, Aronson & King, 2008). In this research, we embrace the definition by Wixom and Watson (2012) who define BI as: "a broad category of technologies, applications, and processes for gathering, storing, accessing, and analyzing data to help

its users to make better decisions". Companies in large proportions are embracing BI. The pressure, which drives companies into embracing BI, is a result of today's competitive business climate. The present business climate drives businesses into being more responsive to the changing business environment. Especially SMEs are vulnerable to the fast changing business conditions (Tutunea, Rus, 2012). In order to be more responsive to this changing business environment, businesses are in need of real-time decision support. This support can be provided by implementing BI systems.

These BI systems provide companies with the benefit of having access to accurate information on demand. This information can for instance be business performance. A company's business performance can, among others, be expressed in the form of for example sales figures, profit margins or geographical data. This information can be used as a source of information when making important decisions. More briefly, these systems provide decision support. A business support system can help in providing faster and more accurate reporting, improved decision-making, improved customer service and an increased revenue (Thompson, 2004).

BI is one of the functionalities that was not affordable for SMEs before the SaaS business model became ubiquitous. On-premises BI solutions were complex and costly. With the SaaS business model, affordable BI solutions became available. Apart from appearing as standalone products, BI functionality was also added to business software such as ERP systems. In ERP systems, BI is often put into practice in the form of dashboards. Dashboards are data visualization tools that provide visual displays of important information, preferably on a single screen so that the information can be processed and explored quickly (Palpanas, Chowdhary, Pinel & Mihaila 2007). A dashboard can for instance help to visualize a trend in the data that is hard to see from the data itself by showing the data in a graph. Dashboards play a big role in a today's business environments. According to Gile (2006) a researcher at Forrester Research, more than 40 percent of the 2,000 largest companies in the world use dashboards.

The most important measure for a software systems success is if the system meets the user's expectations (Cheng, Atlee, 2007). In order to discover what a systems purpose is requirements engineering is used (Nuseibeh, Easterbrook, 2000). Especially in dashboards, the information displayed on the dashboard should be tailored to its user's requirements (Frank, Heise & Kattenstroth, 2009). The data used to create the dashboard can be tailored, but also the way in which it is presented can be tailored. To illustrate this, you can think of a small web shop ran by one person on the one hand and a big oil magnate that operates on a worldwide scale on the other hand. The big multinational is probably interested in sales figures per sales region. For instance, sales in Europe, sales in the USA and sales in Asia. Whilst the owner of the web shop only needs the company wide sales figures, since he is probably not selling enough products to be interested in geographical data. This example is an obvious one, but it illustrates the difference in information needs based on the type of company.

3.5 Customization & Configuration

As seen in the previous sections, ERP systems, Dashboards, and multi-tenant software have to be tailored to its end-users. Tailoring is needed in order to meet the needs of a large number of different customers (Summers, 2001). There are two approaches to reaching a tailored SaaS product, *configuration* and *customization* (Guo, Sun, Jiang, Huang, Gao, Wang, 2008).

Customization involves changing the software products source code. "A customized feature is a feature that has been tailored to fit the nee ds of a specific user" (Jansen, Houben & Brinkkemper, 2010). SaaS providers usually do not provide customization because customization is usually much more costly than configuration (Guo, et al., 2008). This is because the SaaS vendor needs to maintain the separate code instance for every tenant instead of maintaining one single code base.

Configuration in contrast to customization does not involve change in the source code of the software product. Configurability has less ability than Customization. It provides users with configurable options using a single code base. This is why configuration is the preferred way of offering tailoring in SaaS products.

In literature, some sources mentioning *the flexibility usability tradeoff* (which is also called the Flexibility/Specialization Principle) are found. This principle explains that if software gains in flexibility it loses in usability (Lidwell, Holden, Butler, 2010; Steinebach, Guhathakurta, 2009). *Usability* is defined by ISO as: "The effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments"

Flexibility can be interpreted in several ways; in this research the assumption by Guo et al. (2008) is followed, that the term flexibility refers to the exact same thing as tailoring. This means that the tradeoff statement can be interpreted as: A higher level of configuration in a software product decreases its usability. This leads to the hypothesis that if a user is provided with more options to configure, there is more functionality to look at which decreases the simplicity of the product. Less simplicity in turn can lead to decreased usability.

3.6 Web Usage Mining

The ultimate goal of any user-adaptive system is to provide users with what they need without them asking for it explicitly. "Web usage mining refers to the automatic discovery and analysis of patterns in clickstreams, user transactions and other associated data collected or generated as a result of user interactions with web resources on one or more websites". (Mobasher, 2007) The goal of web usage mining is to capture, model, and analyze the behavioral patterns and profiles of users interacting with a website or web application. The discovered patterns are usually represented as collections of pages, objects, or resources that are frequently accessed or used by groups of users with common needs or interests (Mobasher, Cooley, Srivastava, 2000). These patterns can be used for multiple purposes and one of them is creating personalized experiences in the web application.

Web usage mining is a sub discipline of data mining. This means the standard datamining steps as proposed by Fayyad, Piatetsky, and Smyth (1996) in the Knowledge Discovery in Databases (KDD) process can be followed whilst performing web usage mining. The KDD process is depicted in figure 6 below.

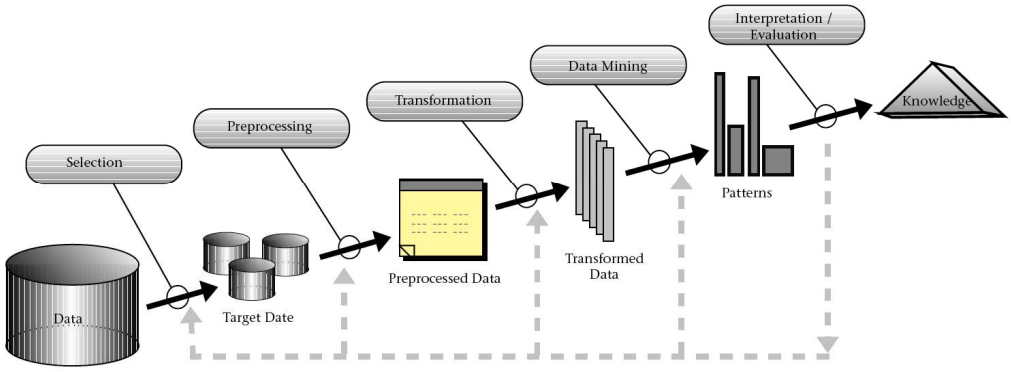

Figure 6. The KDD process as proposed by Fayyad, Piatesky, and Smyth (1996)

Pachidi, Spruit, and van de Weerd (2014), propose a detailed usage mining method as depicted in figure 7 below. This method adds a step before data selection called data understanding. This step is concerned with gaining an understanding of all the available data, and what this data can be used for. The next step is a combination of the first three steps of the KDD process, Pachidi et al. call the combination of data selection, data preprocessing and data transformation 'Data preparation'. The data mining step is split into two separate steps, namely exploratory analysis, and the in depth data mining step.

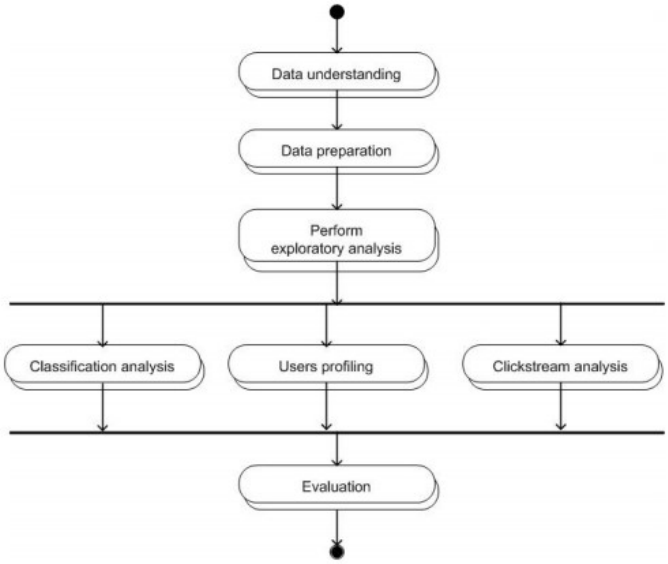

Figure 7. Usage mining method (Pachidi et al. (2014).

Liu (2007), Mobasher (2007), and Srivastava, Cooley, Deshpande, & Tan (2000), combine the first three steps in order to simplify the process to three main steps. These three main steps are: Data gathering and pre-processing, pattern discovery, and pattern analysis. These steps are depicted in figure 8 below. Altough the method by Pachidi et al. (2014) is more detailed it is similar to the steps as proposed by Srivastava (2000). In order to explain the web usage mining steps we follow the steps as proposed by Srivastava (2000) because the steps are more abstract and provide more room to explain the general concepts.

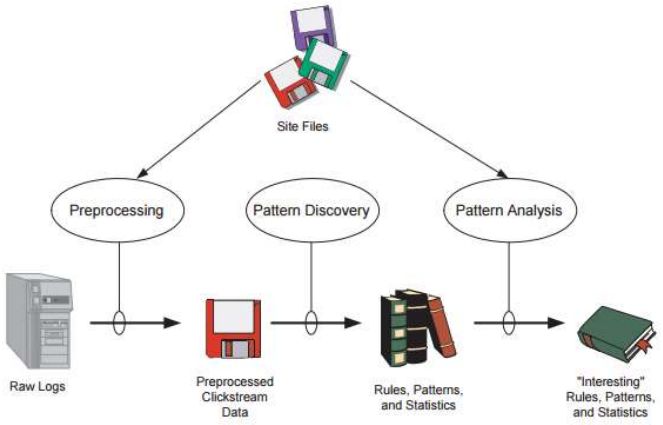

Figure 8. Web usage mining (Srivastava,2000)

3.6.1 Data gathering

The first step of web-usage mining is concerned with gathering the data that is going to be subject of the web usage mining and the pre-processing of this data. The data is typically gathered from multiple sources. The primary and most important type of data in web usage mining is usage data, without usage data there is no point in trying to mine web usage. However, usage data might not be enough for the data mining process. This can be solved by enhancing the usage data with other sources of data. The main categories of other data sources are: Content data, structure data and user data (Mobasher, 2007). The different types of data are explained briefly below.

Usage data is typically gathered from an activity log. An activity log stores the history of user transactions with a website/application. It usually contains information about what pages are being requested, the users requesting the pages, and a timestamp of when the pages are requested.

Content data is data about the content that is on the resources documented in the activity log. This information usually contains text or multimedia files such as images or videos.

Structure data is data that represents the application/website designer's view of the structure of the different resources in the usage data. This is usually represented in HTML and XML documents that contain the link structure of the application. Think of for example a menu structure XML file.

User data is extra data on users that is not stored in the activity log. This can be demographic data about the user, but it can also contain information about for instance the payment plan the user has at the moment.

3.6.2 Data Pre-processing

After gathering the necessary data from the available data sources, the raw-data has to be pre-processed. Usually this consists of data fusion, integration and cleaning, page view identification, user identification, sessionisation, and path completion. (Liu 2007)

Data fusion is needed when the usage data is stored on multiple locations. For instance, in big web applications the usage data is stored on multiple servers in order to reduce server load. Data fusion is the process of merging the data of the different sources of usage data together.

Data Integration is necessary if besides usage data other types of data are gathered. This can for instance be demographic user data, or semantic information on the webpages' contents or structure. These data sources need to be linked to the usage data so it can be analyzed together.

Data Cleaning is the process of removing irrelevant data from the data source. In the case of usage data this can for instance be page views that are not expressing any useful information for the analysis that will be performed.

Page view identification is the process of linking activity log entries to page views. This is important when an action in the activity log may represent multiple page views. This can be the case when a page is dynamic; the user can perform multiple actions on a page that are logged in the activity log as one action. This can also happen the other way around, in the case of websites that use multiple frames to show content multiple actions in the activity log may represent only one page view.

User identification is necessary when a website does not have an authentication mechanism. If a user does not have to login when using the website it is hard to identify the same user over multiple sessions. An approach that is used a lot for user identification is the use of client-side cookies.

Sessionization is the activity of identifying separate sessions of users on a website. This is done to find out what the sequence of page views of a user is during one visit to the website. If there is no authentication mechanism in place sessionization is usually performed by applying heuristics to the data. Heuristics that can be applied to usage data in order to recognize separate sessions can for instance be based on time. An example is that the time difference between two page requests may not be longer than 5 minutes. If the time between two consecutive page requests is longer than 5 minutes they belong to a different session.

Path completion can be necessary if the activity log is not complete. Depending on how the activity log is maintained certain page requests may not be stored in the activity log because of caching. This results in missing activity log entrances, in order to complete the user paths by adding these missing these missing entries heuristics can be applied. A heuristic that has knowledge on the structure of the website can infer if there is a missing page between two page requests.

3.6.3 Pattern discovery

The pattern discovery step is the step in which a data mining method or technique is performed on the data sources gathered and pre-processed in the previous step. Mobasher (2007) names three general ways of data mining for personalization. These are clustering, association discovery, and sequential modeling. Srivastava et al. (2000) in addition to these three techniques propose three more techniques. These techniques are Statistical analysis, classification and dependency modeling. Statistical analysis is concerned with descriptive analyses such as frequency, mean and median, we discard this as a pattern discovery technique because it is too general. Classification is only suitable for research in which predefined categories are available; this is not the case in this research. Dependency modelling on usage transactions is essentially association rule mining. Therefore, this section briefly explains the three techniques as mentioned by Mobasher (2007).

Association Rule Mining

Association rule mining is a data mining technique for discovering regularities between items in a dataset. A widely used example to illustrate association rule mining is that of shopping basket analysis in a web shop. An example of an association rule in this scenario is the following: “Users that buy product A and B also buy product C”. More formally, this rule can be expressed as A, B => C. An association rule consists of an antecedent and a consequent. In the previous example “A, B” is the antecedent and “C” is the consequent. In order to select interesting rules out of the full set of rules two constraints called support and confidence can be added (Mobasher et al. 2000). These measures express how valuable a rule is. Support gives the frequency of the association rule occurring in the data set. In the web shop basket example, this can be added to the rule by stating 50% of the transactions in the web shop consist of users buying product A and B. Confidence gives the probability that a rule holds. So in our example this can be added to the rule by stating for instance: “90% of the users that buy product A and B also buy product C”. This means that we can be 90% sure that if a user buys product A and B he will also buy product C.

Sequential modeling

Sequential pattern mining is a technique that, in the web-mining context, can find different paths users follow between pages of a website/web application. The sequential patterns represent the order in which pages are frequently viewed by multiple users.

Another term used for sequential modeling is clickstream mining (Pachidi et al, 2014). The main difference with association discovery is that sequential modeling takes the order of actions into account, where association discovery does not do this.

A method of sequential modelling is process mining. Process mining is a research field that can be positioned between machine learning & data mining on one side and process models on the other side. The goal of process mining is to use event data to extract process related information from an information system. (van der Aalst, 2011). Process mining focusses on all activities that use event data to extract process information. The goal is to discover or generate a process model by analyzing recorded events in an information system (van der Aalst, 2011). This can only be done when assuming that each event refers to a well-defined step in a process, and each event refers to a case (van Dongen, de Medeiros, Verbeek, & Weijters, 2005). In addition, events can have an actor that is responsible for executing the process step, and can have timestamps. Using these process models, it is possible to check conformance and recommend process redesigns. It is a way to analyze the complex processes and massive amounts of data that underlie these processes (van der Aalst, 2011).

For organizations, it is important to use event data in a meaningful way. Meaningful depends on the purpose, but some examples are the identification of bottlenecks, providing insights, or streamline processes (van der Aalst et al., 2012). Process mining activities can be categorized in several types, known as discovery, conformance, and enhancement (de Medeiros, Guzzo, Greco, van der Aalst, & Weijters, 2008; Song, Günther, & van der Aalst, 2009; van der Aalst, 2011). Discovery techniques take an event log, analyze it, and produce a process model without using any additional information (van der Aalst, 2011). Conformance entails the use of an existing process model, and compare this model with an event log of the same process. This technique can be used to check if the model is in line with the real process, and the other way around. The last type is known as enhancement, which focusses on the extension or improvement of an existing process model by using actual process information from an event log.

Clustering

Clustering is a broad term that can be applied in various ways. The ultimate goal of clustering is always to group a set of objects in a way that objects in the same cluster are more similar to each other than objects in another cluster.

An example of a clustering method is Formal Concept Analysis (FCA). FCA is a hierarchical clustering method. It is also described as a mathematical theory for creating concepts and concept hierarchies (Stumme, 2009). A concept hierarchy can be formed from a dataset by using FCA. In order to perform FCA a formal context is needed. A formal context consists of a set of objects and a set of attributes. Ganter, Stumme & Wille (2002) define a formal context as:

> *"A triple K:= (G, M, I), where G is a set whose elements are called objects, M is a set whose elements are called attributes, and I is a binary relation between G and M ($I \subseteq G \times M$). $(g, m) \in I$ is read 'object g has attribute m'."*

The formal context can be depicted in table form; the cells in the table contain ones and zeroes depending on whether an object has or does not have a certain attribute (Belohlavek, 2011). An example of a formal context is depicted in table 2. The goal of applying FCA is to create a formal ontology; this ontology can be created by applying FCA on the created formal context.

Table 2. Visual representation of an example formal context.

	Descriptor 1	Descriptor 2	Descriptor 3	Descriptor X
Object 1	1	0	1	0
Object 2	1	0	1	0
Object 3	0	1	1	0
Object N	0	0	0	0

Applying FCA to the formal context results in a set of formal concepts. For example, in table 2 object 1 and object 2 form a formal concept together because these are the only objects with the exact same descriptors. Ganter and Stumme (2002) define a formal concept as follows:

"For $A \subseteq G$, let

$$A' := \{m \in M \mid \forall g \in A: (g, m) \in I\}$$

and, for $B \subseteq M$, let

$$B' := \{g \in G \mid \forall m \in B: (g, m) \in I\}:$$

A (formal) concept of a formal context (G, M, I) is a pair (A, B) with $A \subseteq G$, $B \subseteq M$, $A' = B$ and $B' = A$. The sets A and B are called the extent and the intent of the formal concept (A, B), respectively. "

From the set of formal concepts, a concept lattice can be formed which is a visual representation of the relations between the formal concepts. The formal definition of a concept lattice by Ganter and Stumme (2002) is:

"The concept lattice of (G, M, I) is the set of all formal concepts of (G, M, I), ordered by the subconcept-superconcept order defined as:

$$(A1, B1) \leq (A2, B2) :\Leftrightarrow A1 \subseteq A2 \ (\Leftrightarrow B1 \supseteq B2)"$$

If the formal concepts for the example formal context in table 2 are calculated, it results in four formal concepts. These formal concepts can be visualized by creating a concept lattice. The concept lattice for the formal concepts calculated from the formal context in table 2 is shown in figure 9 below. Each circle in the lattice represents a formal concept. In this case, there are four different concepts. The formal concept on the top contains the objects that have the attribute "Descriptor 3" which in this case is not the case for any object, so this concept is empty. The bottom concept contains the objects that have all attributes, which again is not true for any object. The other two concepts are the concept {Descriptor 1, Descriptor 3} containing Object 1 and Object 2 and {Descriptor 2, Descriptor 3} containing Object 3.

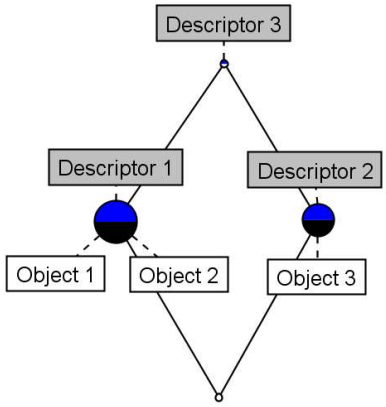

Figure 9. Concept lattice of the formal context in table 2.

3.6.4 Pattern Analysis

The final stage of web usage mining is pattern analysis. The goal of pattern analysis is to interpret the output of the pattern discovery stage. By interpreting the output of the pattern discovery stage useless patterns can be discarded. The resulting, useful, patterns can in turn be applied in order to satisfy the goals of the web usage-mining project. This can for instance be the restructuring of a websites' menu structure, or as in this research for personalization purposes. Depending on these project goals pattern analysis can be applied in a large number of ways.

4. Model Development - Data Gathering and Pre-processing

For the development of the model the web usage mining method, as described in section 3.6 of this document, has been applied. In the coming three chapters the process followed in this research is explained based on the four main web usage-mining steps described in section 3.6. These four main steps are: Data gathering, Data preprocessing, Pattern discovery, and Pattern analysis. A visual representation of these steps can be seen in figure 10. Data gathering and preprocessing are discussed in respectively section 4.1, and section 4.2. The methods considered for pattern discovery are discussed in chapter 5. In chapter 6 the technique used for pattern discovery is described in detail, this leads to a method that can be applied in other ERP systems in order to find different usage patterns of users. In chapter 7 Pattern analysis is discussed. The pattern discovery activities lead to an extended version of the PDD presented in chapter 6. The PDD in chapter 7 extends the previous PDD by adding steps that are specific for the generation of dashboards based on usage patterns. This PDD combines the web usage-mining steps that have been performed, from Data Gathering to Pattern Analysis.

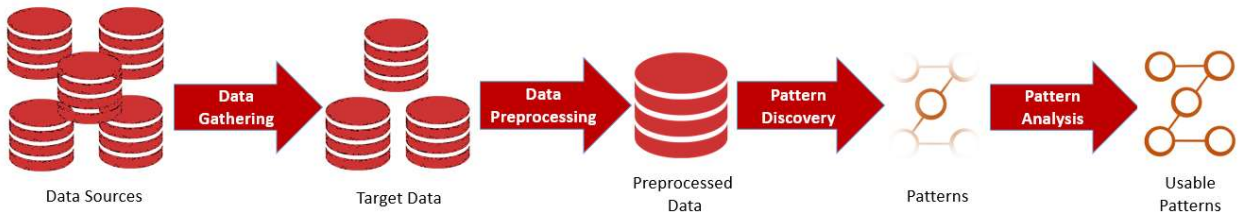

Figure 10. The web usage mining process

4.1 Data Gathering

For the development of the model, all available data sources available at the case study company have been gathered. The case study company provides access to the main types of data sources, namely: Usage data, Content data, Structure data and User data.

4.1.1 Usage Data

The usage data has been retrieved from the activity log of a big SaaS ERP system. The activity log contains usage data of the 250,000 customers of the ERP software product starting from November 2015. The database is still being expanded on a daily basis with around 12,000,000 new entries a day. An example of how a data entry in this database may look is given in figure 11 below. This shows that the author of this thesis on the third of March 2016 at 10:07 looked into the activity log (LgsActivitylog.aspx). This example is used in order not to reveal any real customer data. The data from the activity log is stored in a slightly different format in a SQL database. The transactions in this database consists of, among others, the following information: An accountID, a userID, an aspx and a timestamp.

Start date + time ▼	Application	User	Company
03-03-2016 10:07:18	LgsActivitylog.aspx	Jasper van de Wiel	Exact Netherlands B.V.

Figure 11. An example data entry in the activity log

4.1.2 User Data

Data on the users of the ERP system has been gathered by querying the contract database at the case study company. This database contains extra information on the customers of the SaaS ERP product. This database has been queried in order to get information on: The size of the company of the user in terms of the number of employees, the sector the company is operating in, the region the user lives in, and the package the company of the user is using. The full list of this data cannot be included in an appendix because of user privacy, a snippet of this information can be found in figure 13 below.

4.1.3 Content Data

The content data has been gathered from the ERP system by scraping the aspx'es of the ERP system for their title element. The scraper used for this process has been developed in python making use of the scrapy framework. Part of the resulting data is shown below in figure 12. The complete list of aspx'es and their title elements can be found in appendix A.

FinYearPeriods.aspx?	Overview: Financial years
FinYearPeriodsGen.aspx?	Create: Financial year
GenTopicChildCheck.aspx?	Template: Check
GenTopicLinkAdm.aspx?	Link existing template
HRMAbsenceGantt.aspx?	Leave registration per employee
HRMAbsencePercentages.aspx?	Absence percentages: Overview
HRMAbsenceRegs.aspx?	Overview: Leave registrations
HRMAbsenceTypesCLA.aspx?	Overview: Types of leave
HRMCostCenters.aspx?	Overview: Cost centres
HRMCostUnits.aspx?	Overview: Cost units
HRMDepartments.aspx?	Overview: Departments
HRMDivCategoryNames.aspx?	Overview: Company - classifications
HRMDivisionNew.aspx?	Company - New
HRMDivisions.aspx?	Overview: Companies
HRMEmplAnniversaries.aspx?	Overview: Anniversaries
HRMEmplBirthdayList.aspx?	Birthday list
HRMEmplContracts.aspx?	Overview: Employment contracts
HRMEmployees.aspx?	Overview: Employees
HRMIllnessProgRecover.aspx?	System Error

Figure 12. A snippet of the aspx'es and their title element

4.1.4 Structure Data

The structure data has also been retrieved by using a scraper. This scraper has also been developed in python, making use of the scrapy framework. For the structure data, the menu structure XML file of the ERP system has been scraped. From this file the menu item title, submenu item title, and the menu section title each aspx belongs to, are gathered. A snippet of the resulting data can be seen in figure 13 below. The complete list of aspx'es and the menu items they belong to can be found in appendix A.

Category	Subcategory	Heading	Aspx
Sales	Sales Orders	Sales invoices	LogEntries.aspx
Sales	Items	Items	InvTransactions.aspx
Sales	Sales Orders	Sales invoices	FinInvoices.aspx
Purchase	Returns	Returns	PurReceiptSelect.aspx
Purchase	Goods receipts	Goods receipts	PurReceipts.aspx
Purchase	Returns	Returns	PurReceiptReturns.aspx
Purchase	Purchase orders	Purchase orders	PurPurchOrdersPrint.aspx
Purchase	Goods receipts	Goods receipts	PurPurchOrderSelect.aspx
Purchase	Purchase orders	Purchase orders	PurPurchOrders.aspx
Purchase	Purchase invoices	Purchase invoices	PurPurchInvoices.aspx
Purchase	Items	Reports	PurPurchAnalysis.aspx
Purchase	Purchase orders	Purchase orders	PurEntryPurchOrder.aspx
Purchase	Purchase orders	Purchase orders	PurBalanceList.aspx
Purchase	Inventory	Overviews	LogItemCostPriceChanges.aspx
Purchase	Warehouse transf...	Warehouse tran...	InvWarehouseTransfers.aspx
Purchase	Warehouse transf...	Warehouse tran...	InvWarehouseTransferDeliveries.aspx
Purchase	Warehouse transf...	Warehouse tran...	InvWarehouseTransfer.aspx
Purchase	Inventory	Stock counts	InvStokCounts.aspx
Purchase	Inventory	Overviews	InvStockValue.aspx
Purchase	Inventory	Overviews	InvStockPositions.aspx
Purchase	Inventory	Stock counts	InvStockCountNew.aspx
Project	Invoices	Invoices	SlsInvoicePlanningAccounts.aspx
Project	Entry	Entry	ProWeekTransactions.aspx
Project	Reports	Reports	ProTimeCostTransactions.aspx
Project	Entry	Entry	ProTimeCostApproval.aspx

Figure 13. Snippet of the combination of aspx'es with structure data

4.2 Data Pre-Processing

After gathering the available data the next step is to pre-process this data. As explained in section 3.6.2, there are multiple pre-processing activities. The techniques performed in this research are data integration and data gathering. The other techniques were not applicable to the available data sources.

4.2.1 Data Integration

The first step performed in order to combine the different data sources is data integration. Data integration has been realized by first loading the data sources into a SQL database. This results in a separate table in the database for each of the data sources. The next step is combining the data sources in various ways and storing these combinations into new tables. The combinations that are made in the data integration process are the following:

The usage data has been combined with the user data. This results in usage data that is enhanced with information on region, package, company sector, and company size. A snippet of this information is shown in figure 14 below. The column 'LatestCommPackage' represents the current package of the user, the Environment column shows what country the user is based in. Moreover, the CompSizeCode column shows the size of the company

LogDateTime	Environment	AccountCode	UserID	LatestCommPackage	App	CompSizeCode	IsAccountant
2016-06-23 11:59:13.930	NL	[redacted]	[redacted]	EOLPremium	MenuPortal.aspx	G	Entrepreneur
2016-06-23 12:14:34.727	NL	[redacted]	[redacted]	EOLBasic	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 12:10:41.737	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 12:13:10.633	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 12:15:15.053	NL	[redacted]	[redacted]	EOLPSAAdvanced	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 12:16:23.383	NL	[redacted]	[redacted]	EOLPSAAdvanced	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 12:10:47.377	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	SysPopupMemo.aspx	G	Entrepreneur
2016-06-23 11:55:45.010	NL	[redacted]	[redacted]	EOLAccountancyPremium	ProTimeCostEntry.aspx	G	Entrepreneur
2016-06-23 11:57:07.493	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProTimeCostEntry.aspx	G	Entrepreneur
2016-06-23 11:55:39.697	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProTimeCostEntry.aspx	G	Entrepreneur
2016-06-23 11:59:19.383	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 12:01:18.307	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 12:09:54.940	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 12:01:23.573	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 11:55:47.870	NL	[redacted]	[redacted]	EOLBasic	ProTimeCostEntry.aspx	G	Entrepreneur
2016-06-23 12:12:48.600	NL	[redacted]	[redacted]	EOLPremium	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 12:12:44.850	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 12:17:20.710	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	G	Entrepreneur
2016-06-23 11:59:05.307	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProTimeCostEntry.aspx	G	Entrepreneur
2016-06-23 12:00:11.040	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	HRMAbsenceRegs.aspx	G	Entrepreneur
2016-06-23 12:09:30.097	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	HRMAbsenceRegs.aspx	G	Entrepreneur
2016-06-20 09:18:30.057	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProProjectCard.aspx	NULL	Entrepreneur
2016-06-20 09:12:58.143	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	SysPopupMemo.aspx	NULL	Entrepreneur
2016-06-20 09:12:40.940	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	MenuPortal.aspx	NULL	Entrepreneur
2016-06-20 09:12:46.393	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-20 09:12:43.017	NL	[redacted]	[redacted]	EOLBasic	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-20 09:12:43.003	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-20 09:12:42.613	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	Exact.Online.Web.Areas.CRM.Controllers.AccountL...	NULL	Entrepreneur
2016-06-20 09:12:43.003	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-20 09:18:15.977	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-20 09:18:08.900	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-16 13:49:05.880	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	SysPopupMemo.aspx	NULL	Entrepreneur
2016-06-16 13:48:50.753	NL	[redacted]	[redacted]	EOLPSPremium	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-16 13:48:50.753	NL	[redacted]	[redacted]	EOLPSPremium	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-16 13:48:50.753	NL	[redacted]	[redacted]	EOLPSPremium	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-16 13:48:57.973	NL	[redacted]	[redacted]	EOLPSPremium	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-16 13:50:24.067	NL	[redacted]	[redacted]	EOLPSPremium	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-13 09:23:35.640	NL	[redacted]	[redacted]	EOLPSPremium	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-13 09:23:25.763	NL	[redacted]	[redacted]	EOLAccountancyBasic	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-13 09:23:21.123	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-08 17:46:49.173	NL	[redacted]	[redacted]	NULL	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-08 17:46:49.173	NL	[redacted]	[redacted]	EOLPREMIUM	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-08 17:46:48.220	NL	[redacted]	[redacted]	EOLBASIC	Exact.Online.Web.Areas.Projects.Controllers.Dashb...	NULL	Entrepreneur
2016-06-08 17:49:49.253	NL	[redacted]	[redacted]	EOLAccountancyAdvanced	ProWeekTransactions.aspx	NULL	Entrepreneur
2016-06-08 17:46:51.783	NL	[redacted]	[redacted]	EOLAccountancyBasic	ProWeekTransactions.aspx	NULL	Accountant

Figure 14. Snippet of the user data table

The next combination of data sources is the combination of usage data with structure data. This results in structure enhanced user transactions. This means that menu items, submenu items and menu headers are added to the user transactions in the usage database. This is called a page keyword table; a snippet of this information is shown in figure 15 below. Not all aspx'es are contained in the menu, this means that a number of usage transactions is lost in this integration because they cannot be enhanced with structure data.

User	App	LogDateTime	Category	Subcategory
[redacted]	FinBalanceSheet.aspx	2016-06-26 21:42:24.980	Financial	G/L Accounts
[redacted]	FinBalanceSheet.aspx	2016-06-26 21:42:21.507	Financial	G/L Accounts
[redacted]	FinBalanceSheet.aspx	2016-06-26 21:42:17.370	Financial	G/L Accounts
[redacted]	FinJournalSelect.aspx	2016-06-26 21:46:09.540	Financial	Entries
[redacted]	QuoEntryQuotation.aspx	2016-06-26 20:36:59.993	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 21:09:49.773	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 21:09:19.380	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 21:12:21.763	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 21:13:02.123	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 23:00:26.273	CRM	Quotations
[redacted]	QuoEntryQuotation.aspx	2016-06-26 23:00:07.620	CRM	Quotations
[redacted]	SlsSalesOrderPicklist.aspx	2016-06-26 23:41:31.737	Sales	Sales Orders
[redacted]	CflImport.aspx	2016-06-26 21:58:09.333	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 23:04:15.350	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 23:33:50.000	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:02:26.767	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:55:59.943	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:58:09.477	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 22:25:11.743	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 23:33:40.373	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 20:52:32.013	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:39:24.367	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 23:04:19.053	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 23:04:18.957	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:39:35.210	Financial	Bank
[redacted]	SlsSalesOrderPicklist.aspx	2016-06-26 23:41:11.347	Sales	Sales Orders
[redacted]	CflImport.aspx	2016-06-26 21:55:59.800	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 20:40:32.703	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:02:26.393	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:02:19.033	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:55:56.380	Financial	Bank
[redacted]	CflImport.aspx	2016-06-26 21:58:04.177	Financial	Bank
[redacted]	FinJournalSelect.aspx	2016-06-26 22:15:55.197	Financial	Entries
[redacted]	FinJournalSelect.aspx	2016-06-26 22:53:30.567	Financial	Entries
[redacted]	FinJournalSelect.aspx	2016-06-26 22:59:24.133	Financial	Entries
[redacted]	FinJournalSelect.aspx	2016-06-26 20:31:37.703	Financial	Entries
[redacted]	FinJournalSelect.aspx	2016-06-26 22:53:36.977	Financial	Entries

Figure 15. Page keyword table

4.2.2 Data Cleaning

After data integration, the data has been cleaned in various ways. The irrelevant users, aspx'es, and menu items, have been removed. The motivation for classifying users, aspx'es, and menu items, as being irrelevant is described below.

The first data to be cleaned is the usage data, gathered from the activity log. It has to be cleaned of actions that are irrelevant. Actions can be irrelevant because of multiple facts. These facts are shown in table 3 below, accompanied with an example and a reason. For every irrelevancy, there are multiple examples, in the table only one example is stated for each reason. A full set of irrelevant aspx'es is provided in appendix B. The aspx'es that are classified as not manual are not included in the appendix, this would unnecessarily clutter the appendix.

Table 3. Examples of irrelevant page views

Irrelevancy	Example	Reason
Express no information	Menuportal.aspx	These aspx'es turn up in the activity log for every user, because the functionality is general system functionality.
Not manual	restapi/entity/crm/accounts	This is a non-manual action performed by an app.
Too general	FinTransactions.aspx	This aspx has multiple functions depending on a parameter that is not stored in the activity log.
Related to Country Specific Legislation	AssCDioxide.aspx	This page is related to the specific Carbon Dioxide tax laws as they are in Belgium.
Related to license/master data or hosting & reseller accounts		These pages are not relevant for regular exact online users
Dashboards, Quickbooks, Reports		Thesepages are removed from the analysis, because these pages are directly related to dashboards.

The next data to be cleaned is the user data, gathered from the contract database. This data consists of a number of items being: The size of the company determined by the number of employees, the sector the company is operating in, the region the user lives in, and the package the company is subscribed to. However, this data has a big number of missing entries. In fact, the data on the company size, and the data on the company sector are so incomplete that both columns have been discarded in the data.

The region information is used to delete users not situated in the Netherlands or Belgium. Users that are not in these countries cannot be visited in order to include them in the case study and are therefore not relevant users to base the model on. Another scoping decision is not to look at all packages offered by the ERP vendor, in this research only the packages for manufacturing, production and professional services companies are taken into account. This means users that have an accountancy subscription are discarded. The ERP vendor also offers educational licenses for high schools and universities, these licenses are discarded because students are not representative for employees of SMEs.

After cleaning the usage and user data, the semantic data also has to be cleaned. The content data consisting of page title elements has some missing data. This means some of the missing, or wrongly scraped page titles have been completed manually. The missing page titles are caused by pages that do not have their title between HTML <title></title> tags, but between for instance <h1></h1> tags or by pages that have no title at all. Pages that expect a parameter when being invoked are the main cause for the wrongly displayed page titles. Because of the way the scraper is programmed it does not add parameters to page requests, this causes the title of the page to be: "System Error".

The structure data has not undergone any data cleaning operations because this data was already in the correct format for analysis. However, some submenu items are not present in the combined dataset of usage and structure data. This is because for some submenu items all related aspx'es have been deleted from the usage data because of reasons stated in table 3 above.

5. Model Development - Methods Considered for Pattern Discovery

After gathering and pre-processing the data, the data-mining task can be performed in order to discover patterns in the data. In this research, the patterns represent different usage patterns expressing the different information needs of users. As explained in section 3.6, Mobasher (2007) names three main methods for data mining on web usage data. For the development of the model, all three of these methods have been considered.

Association rule mining has been discarded in early stages of the research. For sequential modelling, process mining is a technique that has been looked at in detail. Eventually clustering has been used in order to develop the model. To be more specific, content enhanced clustering has been applied, using Formal concept analysis (FCA) as the clustering technique.

In this chapter, it is shortly elaborated why association rule mining has been discarded as methods for developing the model. Then process mining, and how it has been attempted to use it in this research, is explained. Lastly clustering is discussed. First, the attempts to apply k-means clustering are discussed briefly, next formal concept analysis and our approach to FCA are explained in detail. This chapter provides an answer to the first sub research question:

SRQ1: How can the different usage patterns of SME employees, while using ERP systems, be identified?

5.1 Association Rule Mining

Association rule mining has been considered as a technique to be used to create a model of different usage patterns. The idea is applying association rule mining on the activity log in order to find sequences of pages regularly followed by different users. Performing association rule mining on the usage transactions results in a large number of association rules with an acceptable confidence value. However, the support for these rules is very low. Table 4. below shows the number of strict association rules for the analysis on 3 abstraction levels of the dataset, along with their average support rates. The low support rate shows that there is little use for these rules in the expression of different usage patterns. Based on this data, it has been decided to look into a more sophisticated technique for data mining that provides better ways of interpreting the results.

Table 4. Summary of the Association Rule Mining Findings

Dataset abstraction level	Number of strict rules	Average support of strict rules
Aspx'es	200844	0.12%
Submenu items	1337	0.32%
Menu items	5	0.09%

5.2 Process mining

Process mining can provide us with an overview of different paths users take in the ERP system. These different paths can possibly represent groups of users with the same behavior and therefore the same information needs. Because the data is gathered from an activity log, all data necessary for processes mining is available. There are timestamps, cases (users) and events (aspx requests). For this research, the discovery technique of process mining is the most appropriate one because there is no existing model to improve, or to compare the event log with.

The two tools that are most commonly used for process mining are ProM and Disco. Because ProM has slightly more functionality this was the first choice of process mining software. However, the software is slow and unstable when confronted with a large dataset as the one used in this research. This is why Disco has been used to perform process mining.

The first step is loading the dataset, consisting of user transactions combined with user data, into Disco. When visualizing this data Disco shows a so called 'spaghetti model'. This complex figure is too overwhelming to provide any value to our research, see figure 16 below. However, the user data gathered from the contract database provides us with means to filter on extra attributes in Disco. The attributes the user data provides are: Region, subscription type, company size, and sector.

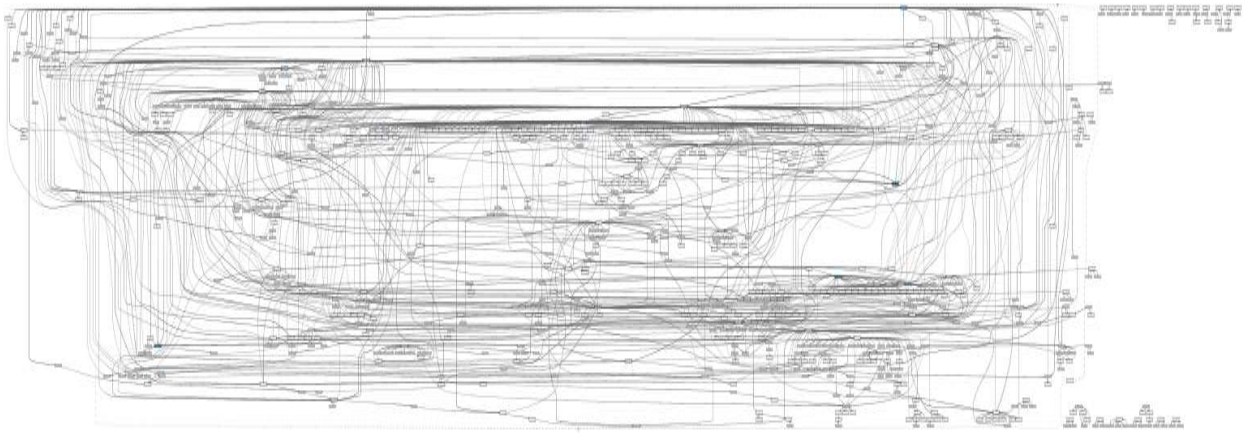

Figure 16. Disco Spaghetti model

Filtering enables looking at smaller parts of the dataset at once. This means the data can be filtered to only show the paths followed by for instance Dutch users with a manufacturing subscription. Applying these filters did simplify the models. One path used by a big group of users could be identified in these filtered models. For example, the path in figure 17 consists of only pages related to time entries. This means the users that follow this path only use the ERP system for time entries. However, other than this one specific path there are still too many paths between the different pages in order to identify different types of customers. There are some frequently followed paths, but these are used by all users, so they do not express any different information needs.This means a model of usage patterns cannot be created applying this method on the usage data combined with user data.

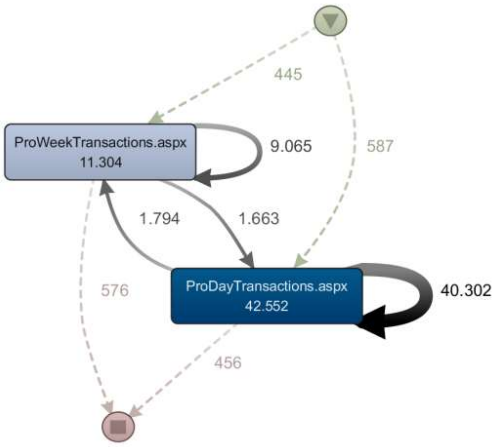

Figure 17. Path followed by time entry users visualized in Disco

However, another combination of data is still available. The usage data combined with the structure data is therefor also analyzed in Disco. Filtering on main menu item results in a clear overview, unfortunately this overview does not express any useful information. Using the submenu items to filter the data again results in a spaghetti model similar to the spaghetti model in figure 15 above. This leads us to the conclusion that process mining is not an appropriate technique to discover usage patterns from usage data.

5.3 Formal Concept Analysis

Formal Concept Analysis is a clustering technique that can help us by providing a concept lattice containing concepts consisting of users visiting the same pages within the ERP system. Users with the same usage patterns could possibly have the same information needs. The data needed to create such a concept lattice is available to us. The descriptors are the aspx'es in the ERP system and the objects are the user IDs from the activity log.

Before looking into formal concept analysis multiple clustering techniques have been explored. These techniques, including a basic k-means clustering approach were not powerful enough to find different usage patterns in our dataset. This section describes two different ways that have been used to apply FCA to the data. It is first explained how the preprocessed data has been transformed in order to be able to apply FCA on it. Then the first method of applying FCA is described, and after that, the second method of applying FCA is explained.

5.3.1 Formal Context Creation

The format of the usage data after data cleaning is not suitable for application of the FCA algorithm. The usage data is in the format of a big csv file with a single user action on every line. Moreover, as can be seen in table 2 above in the explanation of FCA, the formal context has to be of another format. The rows need to represent a user and the columns need to represent a certain aspx. The values in the table should be of a binary value representing if a user requested a certain aspx or not. Online, two conversion tools can be found that are developed in order to convert data into a format suitable for FCA. These tools are called FcaStone and FcaBedrock. Unfortunately, both tools froze when confronted with our dataset. This is most likely caused because the dataset contains too many rows for the software to handle. In order to get the data into the right format some data transformation operations have to be applied, including a pivot table.

Microsoft Office Excel is not a suitable solution in order to perform these data transformations. At least not on a dataset the size of the one in this research. The worksheet limit of an Excel file is 1,048,576 rows by 16,384 columns, and the data set in this research exceeds that number by far. This is why a python script has been created that performs a pivot table on the output file. The python script makes use of the pandas library, this is a library specifically created for the purpose of data analysis activities. With this script, the dataset has been transformed into the right format. Part of the formal context can be seen in figure 18.

UserID	AccGLJournalOverview.aspx	CflExpensesReceiptsEntry.aspx	CflImport.aspx	CflMatch.aspx	CflPayments.aspx
00F18	0	0	0	0	0
03F35	0	1	0	0	0
049F8C	0	0	0	1	0
08C06	1	0	0	0	0
0AF2C	0	0	0	0	0
0BC8A	0	0	0	0	1
0BCB4	0	0	0	0	0
0C944	0	0	1	0	0
0DD95	0	0	0	0	0
0F5BA	0	0	0	0	0
11178	0	0	0	0	0
12917	0	0	0	0	0
181AE	0	0	0	0	0
19B02	0	1	0	0	0
19D76	0	0	0	0	0
1B2B9	0	0	0	0	0
1BD21	1	0	0	0	0
1C0D6	0	0	0	1	0
1C4AE	0	0	0	0	0
1DD3A	0	0	0	0	0

Figure 18. Part of the formal context

5.3.2 Formal Concept Analysis

For the execution of the FCA algorithm, a number of tools has been explored and tried out. The first two options explored are java tools called ConExp and In-Close. Both tools are not able to handle the dataset and freeze when trying to import the data. The next option looked into is a python script called concepts.py. This script behaves more stable than the java tools when confronted with a small sample of the dataset. This script looked promising, also because the output format can be imported into graphviz, which is a nice tool to visualize the concept lattice. However, after running the script on the complete dataset for over 200 hours, it still did no output anything. In the documentation of the script the authors state “If you want to apply FCA to bigger data sets, you might want to consider other implementations based on more sophisticated algorithms like In-Close or FcbO” As stated above In-Close has already been tested, and did not work.

Logically FcbO was the next option. FcbO did indeed work with the dataset, and after 36 hours of running this script it delivered an enormous output. Unfortunately, this output file, in combination with the form of the output, is so complex it is impossible to make any useful observations from the data. This leads to the conclusion that performing FCA on only the usage data is not feasible. The reason for this is that the technique is computationally infeasible on such a big and diverse dataset.

Table 4. Different FCA tools and their performance on our dataset

	Import	Output	Usable Output
ConExp	No	No	No
In-close	No	No	No
Concepts.py	Yes	Yes	No
FcbO	Yes	Yes	No

5.3.3 Enhanced Formal Concept Analysis

However, the data gathered in the data gathering and pre-processing step can help enhance the usage data. The usage data can be combined with one, or multiple, of the other data sources. This idea is supported by a number of scientific papers that suggest adding semantic knowledge about page views to the data before performing any kind of data analysis, or to be more specific: Before clustering. This technique is called content-enhanced clustering.

Dai and Mobasher (2003) propose to use information associated with items on a website/application as a source of semantic knowledge. As explained in section 4.1 this data can be of two forms. On the one hand, there is content data containing for example text snippets from the pages. On the other hand, there is structure data that represents the application designer's view of the structure of the different resources in the usage data. It has been decided to use the usage data combined with the structure data in order to create different abstraction levels in the data.

In order to do this, first a formal context has been created containing user actions on a menu item level. Next, a formal context containing user actions on a submenu item level has been created. The tables that result from these data integration actions are called content enhanced transaction matrices. The combination of the usage data and the structure data into a content enhanced transaction matrix enables us to look at the data from different levels of detail. This means separate sub-sets of the dataset can be analyzed one by one. After some testing, we managed to create a model of usage patterns using formal concept analysis on subsets of the data making use of structure data to create these subsets. The development of the model is described in the next chapter.

UserID	Accounts	Bank	Budgets	Campaigns	Contact persons	Cost centres	Costs units	Currency	Entries	Entry	G/L Accounts	Goods receipts
FFFD57AC-...	1	1	0	0	1	0	0	0	1	0	0	1
FFFC78FE-...	1	1	0	0	1	0	0	0	1	0	1	0
FFFC76FC-...	1	0	0	0	1	0	0	0	1	0	0	1
FFEA872F-...	1	1	0	0	0	0	0	0	1	1	1	0
FFE51404-...	1	1	0	0	0	0	0	0	1	0	1	1
FFCF8041-...	1	0	0	0	0	0	0	0	0	0	1	1
FFCB9BCC-...	1	0	0	0	1	0	0	0	1	0	1	0
FF789586-...	1	1	0	0	0	0	0	0	1	1	1	0
FF736103-...	1	1	0	0	0	0	0	0	1	1	1	0
FF6FD805-...	1	1	0	0	1	1	0	0	1	0	1	1
FF58056B-...	1	1	0	0	0	0	0	0	1	0	0	0
FF44E05D-...	1	0	0	0	0	0	0	0	1	0	1	0
FF418879-...	1	1	0	0	1	0	0	0	1	0	1	0
FF11F397-...	1	1	0	0	0	0	0	0	1	0	0	1
FF0BDF45-...	1	1	0	0	0	0	0	0	1	0	1	0
FEF7382B-...	1	1	0	0	0	0	0	0	1	0	0	1
FEDD0723-...	1	0	0	0	0	0	0	0	1	0	1	1
FE8DAB06-...	1	1	0	0	0	0	0	0	1	0	1	0
FE70719B-...	1	0	0	0	1	0	0	0	1	0	0	0
FE6E9A79-...	1	1	0	0	0	0	0	0	1	0	1	0
FE43892E-...	1	0	0	0	0	0	1	1	1	0	1	0
FE2EF1C2-...	1	0	0	0	1	0	0	0	1	0	0	0
FE128441-...	1	1	0	0	0	0	0	0	1	0	0	0
FE05239D-...	1	0	0	0	0	0	0	0	1	1	1	0
FDF90606-...	1	1	0	0	1	0	0	0	1	0	1	0
FDD6425D-...	1	1	0	0	0	0	0	0	1	0	1	0
FDD09C74-...	1	0	0	0	0	0	0	0	0	0	1	1
FDC269F2-...	1	1	0	0	0	0	0	0	1	0	1	0
FD4965F1-...	1	0	0	1	1	0	0	0	1	0	1	1
FD3DD0E7-...	1	0	0	0	1	0	0	0	1	0	0	1
FD360665-...	1	1	0	0	1	1	1	0	1	0	1	0
FD35151F-...	1	1	0	0	1	0	0	0	1	0	1	1
FD2FFA0F-...	1	1	0	0	0	0	0	0	1	0	0	0
FD2C8741-...	1	1	0	0	1	0	0	0	1	0	1	0
FD2BF93C-...	1	0	0	0	0	0	0	0	1	0	0	0
FD2483A3-...	1	1	0	0	0	0	0	0	1	0	1	0

Figure 19. Part of the content-enhanced transaction matrix

5.4 Chapter Conclusion

After looking into different techniques of pattern discovery, it can be concluded that content enhanced formal context analysis is a technique that can be used to prove that there are different usage patterns. This means this technique can be used in order to find different usage patterns. These usage patterns can be discovered from a combination of the usage and structure data collected in the data collection stage. From the two data sources a content-enhanced transaction matrix can be created, on which FCA can be performed. This answers the first sub-research question of this research: "How can the different usage patterns of SME employees while using ERP systems be identified?" In the next chapter, content enhanced FCA is used in order to discover usage patterns.

Table 5. Summary of the different mining techniques

Technique	Ease of Use	Interpretability of patterns	Interpretability of patterns after enhancing the data
Association Rule Mining	***	*	*
Process Mining	***	*	**
Formal Context Analysis	**	*	***

6. Model Development – Pattern Discovery

After a long search for a technique that can discover different usage patterns in the data gathered from the ERP system, the final method used for pattern discovery is content enhanced clustering, using formal concept analysis as the clustering technique. In order to discover and visualize different usage patterns, FCA is performed on the content enhanced dataset that has been discussed in the previous chapter. This means FCA is performed on sub-sets of the dataset. These subsets are created based on different levels of abstraction. These abstractions levels are based on the structure data scraped from the ERP system. The context data scraped from the ERP system is used later on in the Pattern Analysis phase of web usage mining. The pattern discovery actions described in this chapter result in a model of different usage patterns. This model answers sub research question 2:

SRQ2: What are the different usage patterns of SME employees while using ERP systems?

The development of the model is the final step of the pattern discovery phase in the web usage mining approach. The next phase in the web usage mining approach is pattern analysis. The pattern analysis performed on the model answers sub research question 2 and is discussed in chapter 7.

6.1 The Dataset

As stated in the previous section a dataset has been created by combining multiple data sources. These data sources are: User transactions gathered from the ERP system's activity log, and context information scraped from the menu structure. The user transactions used in the analysis span one month (4 weeks containing 20 working days and 4 weekends). Table 6 shows that the data contains usage statistics of 21.637 unique users with a total number of actions of over 15.5 million. The data of a month has been used in order to get a complete picture of the actions users perform, without the dataset getting too big to perform an analysis on it. A month is suitable because some actions only happen at the beginning of a month and some actions only happen at the end of a month. Another reason why this data is representable is that the number of actions in the timespan of our dataset is almost equal to the average of the actions per month over a whole year. As can be seen in table 7 below, the average number of actions per month is 289 million. Note that these numbers are much higher than the number of actions in our dataset because they include non-manual actions.

Table 6. Information on the used dataset

Timespan	Unique Users	Total Actions
30-05-2016 to 26-06-2016	21.637	15.542.569

The context information scraped from the menu structure of the ERP system consists of a hierarchy of main-menu items that have submenu items that in their turn have aspx'es. Part of this structure is shown in figure 14 in chapter 4. The combination of the usage data and the context information is called a page-keyword table; part of this page-keyword table is shown in figure 14 in chapter 4. The hierarchy in this page-keyword table makes it possible for us to perform FCA on multiple abstraction levels. Which is explained below in section 6.2.

Table 7. Overview of total number of actions per month

Year – Month	Total number of actions (Millions)
2015-11	241
2015-12	242
2016-01	322
2016-02	277
2016-03	287
2016-04	326
2016-05	265
2016-06	284
2016-07	309
2016-08	254
2016-09	296
2016-10	367
Average	**289**

6.2 Performing Content Enhanced FCA

The highest level of abstraction, that provides the least detailed view, is grouping the user's page views by the menu category each page belongs to. Performing FCA on this abstraction of the dataset provides us with the concept lattice as shown in figure 20. This concept lattice is created by performing FCA on a formal context created by using user IDs as objects and main-menu items as attributes. The main menu items are 'Purchase', 'Sales', 'Person', 'Project', 'CRM', 'Financial', and 'Manufacturing'.

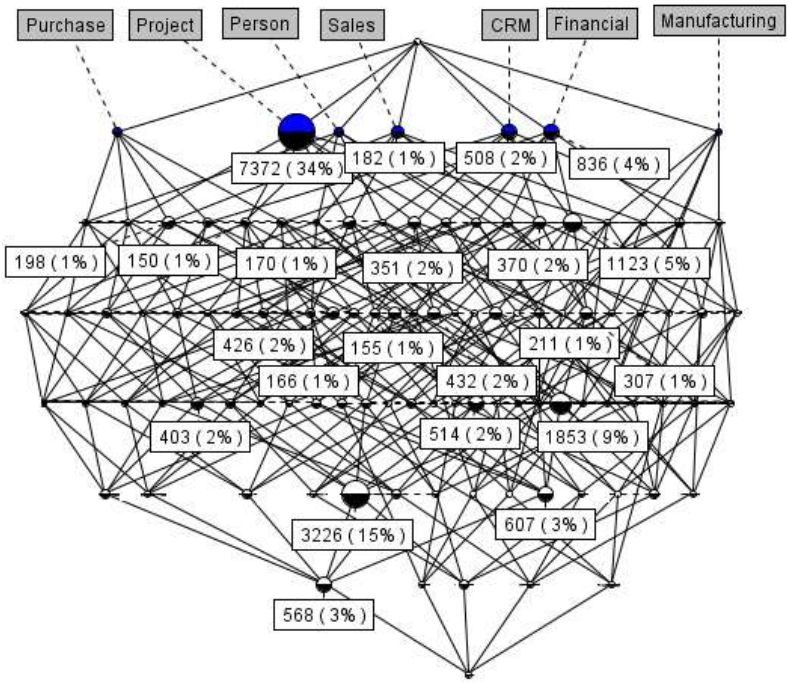

Figure 20. Concept lattice of page views grouped on main-menu item.

This concept lattice shows groups of users that visit pages with the same main menu item related to them. The notion of a concept lattice has been explained in chapter 3. In this lattice, the objects are users, represented by user IDs, and the attributes are main menu items. A new concept in this lattice are the labels that do not state the objects, but state a number and a percentage. This label is explained by looking at the biggest node in this lattice. This is the second node from the left, on top of the diagram. As can be seen this node groups the users that only view pages under the 'Project' menu item. This node is accompanied by a label stating "7372 (34%)". '7372' stands for the number of users that are in this node, so 7372 users only view pages that are grouped under the 'Project' main menu item. The percentage represents the percentage of users that are in this concept related to the total number of users in the dataset.

Not all the concepts in figure 20 are accompanied by a label. This is because the labels of concepts with less than one percent of the users in them have been removed, in order to keep the image clear. This means the 22 biggest concepts of this lattice are shown and discussed in this chapter (The concepts accompanied with a label in figure 20). This accounts for 95% of the users of the ERP system and their actions in the month July (30-05-2016 to 26-06-2016). The other concepts also express different usage patterns, but for the explanation of our pattern discovery technique and the model development, the 21 biggest concepts give a clearer overview.

For clarification table 8 below numbers the 22 concepts. It also provides information on the attributes in each concept, the number of users in the concept, and the corresponding percentage in terms of the total number of users. The newly introduced concepts and sub-concepts are clarified in similar tables later on. Each (sub-) concept has a unique number, which is used to refer to the concept throughout the complete document. The percentage in the last column is always expressing the percentage of users in relation to the total number of users in the (sub-)concepts' super-concept. Not every concept is shown in this document. All concepts, lattices and explanatory tables of the model can be found in the appendices of this document.

Table 8. The 22 biggest concepts on Category level

Concept #	Attributes	Users	Percentage
1	Project	7372	34%
2	CRM, Financial, Project ,Purchase, Sales	3226	15%
3	CRM, Financial, Project, Sales	1853	9%
4	CRM, Financial	1123	5%
5	Financial	836	4%
6	CRM, Financial, Person, Project, Sales	607	3%
7	CRM, Financial, Person, Project, Purchase, Sales	568	3%
8	CRM, Financial, Purchase, Sales	514	2%
9	CRM	508	2%
10	CRM, Purchase, Sales	432	2%
11	CRM, Project, Sales	426	2%
12	CRM, Sales	403	2%
13	CRM, Project, Purchase, Sales	370	2%
14	Person, Project	351	2%
15	CRM, Financial, Sales	307	1%
16	CRM, Financial, Project	211	1%
17	Purchase,Sales	198	1%
18	Sales	182	1%
19	Project, Sales	170	1%
20	CRM, Financial, Purchase	166	1%
21	Financial, Project, Sales	155	1%
22	Person, Project	150	1%

The fact that clusters of users that have similar usage patterns are expressed by the output of the formal concept analysis, means that FCA is a suitable approach to discover different usage patterns. However, the distinction of different usage patterns based on main menu items is not specific enough to link information needs to them in the pattern analysis step. Therefore, the next step is a drill down on these 22 concepts in order to get more detailed sub-concepts out of them.

The lowest level of abstraction, and therefore the highest level of detail, that can be reached are the single aspx'es. The abstraction level between the main menu categories and the aspx'es is looking at the users page views on the submenu item level. For the development of the model, the submenu item abstraction level has been considered. This level of abstraction has been chosen because the category abstraction level is too abstract, and therefore does not express enough information about different information needs of usage patterns. The aspx level on the other hand is too detailed to base usage patterns on.

An example of this drilling down is shown below in figure 21. Here the actions of the users in concept 4 as shown in table 8 have been enhanced with submenu item data instead of main-menu item data. The CRM menu item has three submenu items. These are 'Accounts', 'Contact Persons', and 'Quotations'. In addition, the Financial menu item has 4 submenu items. These are ''Bank', 'Currency', 'Entries', and 'G/L Accounts'. In reality, each main menu item has more submenu items but these have been deleted in the data-cleaning step as explained in chapter 5. Again, the labels of concepts with less than 1% of the total number of users in this lattice have been deleted from the image to get a clearer image. Again, a table of the biggest concepts in this lattice has been created. This table is shown below in table 9.

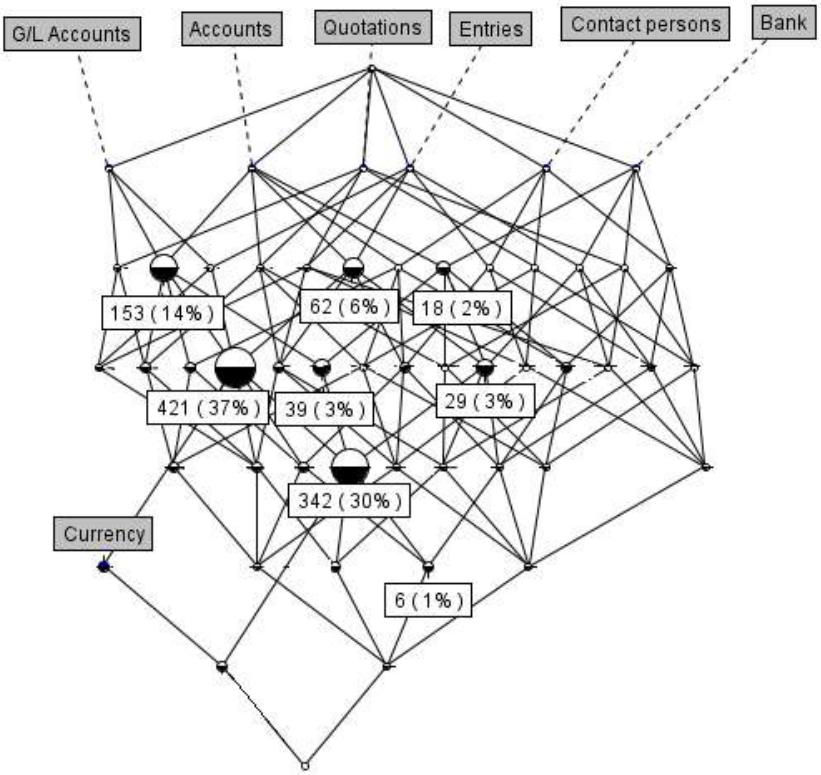

Figure 21. Concept lattice of concept 4.

Table 9. Sub-concepts of concept 4

Concept #	Attributes	Users	Percentage
4.1	Accounts, Entries, G/L Accounts	421	37%
4.2	Accounts, Bank, Entries, G/L Accounts	342	30%
4.3	Accounts, G/L Accounts	153	14%
4.4	Accounts, Entries	62	6%
4.5	Accounts, Bank, G/L Accounts	39	3%
4.6	Accounts, Bank, Entries	29	3%
4.7	Accounts, Bank	18	2%
4.8	Accounts, Bank, Contact Persons, Entries, G/L Accounts	6	1%

Drilling down one level deeper to aspx level gives us the concept lattice in figure 22. This lattice is already becoming hard, or even impossible, to read for the human eye. This level of abstraction also starts getting computationally hard, and does not express useful information. Because the submenu items already express a diverse set of usage patterns, it has been decided to base the model on the submenu item abstraction level. A concept lattice of the concepts listed in table 8 has been created on submenu item abstraction level. The concepts in these lattices have been combined together in a large table file in order to create a textual model. This led to ~1600 different usage patterns, these usage patterns are numbered in the same way the concepts have been numbered in table 8. In table 10 below, the example usage patterns 4.1, 4.2 and 4.3 are shown.

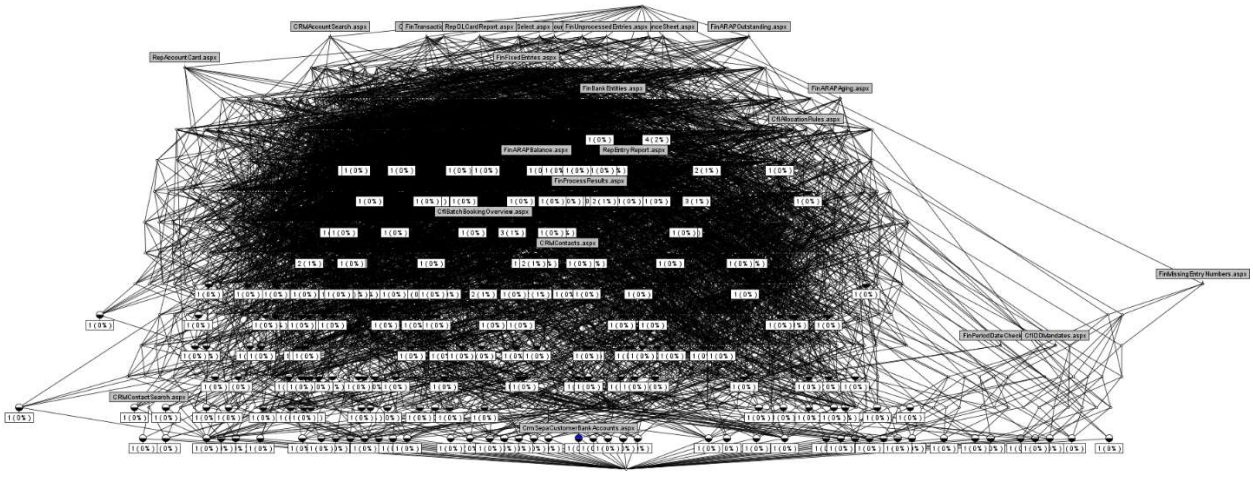

Figure 22. Concept 4, on page view abstraction level

Naming the patterns mentioned above makes no sense because there is no ground to base the naming on. A large portion of the usage pattern only differ from each other in only one or two submenu items. In theory there can be 2^19=524,288 different usage patterns, because there are 19 submenu items that are taken into account in pattern discovery. In addition, each submenu item can have two values, namely '1' (visited) or '0' (not visited). Therefore, it is decided not to use the textual model, as the final model created in the pattern discovery step. Instead, the model is expressed by visualizing the method followed to group the user behavior in to usage patterns in a step-by-step PDD. The PDD is accompanied by an explanation of each of the steps and its deliverable.

Table 10. Example of three usage patterns.

Subcategories	User Type 4.1	User Type 4.2	User Type 4.3	
Accounts		1	1	1
Contact Persons		0	0	0
Quotations		0	0	0
Bank		0	1	0
Currency		0	0	0
Entries		1	1	0
G/L Accounts		1	1	1
Employees		0	0	0
Payroll		0	0	0
Entry		0	0	0
Invoices		0	0	0
Projects		0	0	0
Planning		0	0	0
Inventory		0	0	0
Purchase Orders		0	0	0
Returns		0	0	0
Items		0	0	0
Sales Orders		0	0	0
Subscriptions		0	0	0

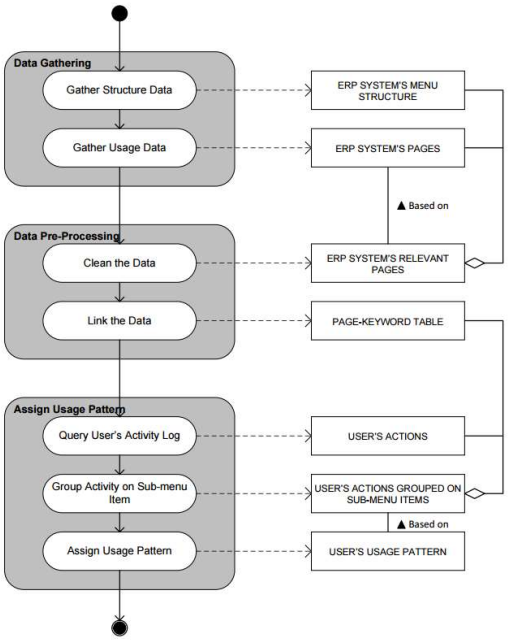

Figure 23. PDD of the usage pattern discovery method

This PDD shows how the model can be used to link a usage pattern to a specific user. The first activity is data gathering. This activity consists of two steps in which the structure data and the usage data are gathered from the ERP system. The structure data is gathered by scraping the menu system of the ERP system. The usage data is gathered from the activity log of the ERP system. The deliverables linked to these two steps are the ERP system's menu structure, and the ERP systems pages. The usage data contains more information, but for the creation of the page-keyword table, only the pages that the ERP system consists of are needed.

The next activity is the data pre-processing, this activity consists of two steps. The first step is data cleaning; in this step the relevant pages of the ERP system are selected from the usage data. What pages are relevant is dependent on what the usage patterns are going to be used for. For example, in this research the login page is discarded because it does not express any value for the creation of personalized dashboards. The next step is linking the cleaned usage data to the ERP system's menu structure so that each relevant page is linked with a menu item and if applicable a submenu item. This results in the Page-Keyword Table. The Page-keyword table can be loaded into a relational database.

The third and last activity is assigning usage patterns to specific users. This activity consists of three steps. First the user's activities in the past 30 days are queried. These are stored in a different table in the relational database that the page-keyword table is also stored in. Next the users' actions table is joined with the page-keyword table. With a count statement on the number of visits to a submenu item. This leads to a view that shows the users' actions grouped on how many times the user visited pages under a certain submenu item. The last step is deducting a usage pattern from this table. In this research, a threshold of five visits per submenu item has been used, to count a submenu item in the usage pattern. The usage pattern consists of a string of 1's and 0's based on what subcategories a user has visited. A one means that pages belonging to this submenu item have been visited more than the threshold value. A zero means pages belonging to this submenu item have been visited less than the threshold value.

The usage pattern string, which is determined by applying the steps of the model, can now be used in order to create personalized information in the system. In this research, the usage pattern is specifically used to generate personalized dashboards. Therefore, information needs are linked to the submenu items. These steps are discussed in chapter 7. In this chapter some more steps, specific for the generation of personalized dashboards are added to the PDD.

6.3 Chapter Conclusion

From this chapter it can be concluded that content enhanced FCA can indeed be used to discover different usage patterns. These usage patterns can be based on the submenu item abstraction level. The concept lattices created using FCA give insights in how the different kinds of usage are distributed in percentages of the total number of users. It can be concluded that there are a big number of different usage patterns that can be generalized in order to group similar users together. However, the high number of different usage patterns, in combination with the slight differences between those users' types makes it impossible to give the usage patterns meaningful names. Also because of the aforementioned high number of different usage patterns, a textual model of these different usage patterns becomes hard to grasp because of its size. It is therefore decided to express the model of different usage patterns in the form of a method. Using this method enables linking a usage pattern to every user of the ERP system by looking at their activity data.

The method expressed by the PDD in figure 23, provides a way to answer our second research question: *"What are the different usage patterns of SME employees while using ERP systems?".* In this document, the question is only answered for the ERP system developed by the case study company of this research. However, with the PDD provided in this chapter the method can be repeated at other ERP software companies as well, in order to discover the different usage patterns of SME employees using that specific ERP system.

7. Model Development – Pattern Analysis

After pattern discovery, as discussed in chapter 6, the next phase in the web usage approach is Pattern Analysis. In this research, the goal of pattern analysis is analyzing how information needs can be linked to the different usage patterns, and linking dashboard components to these information needs. The result of the pattern analysis stage answers the third sub research question:

> ***SRQ3**: How do the usage patterns express different information needs of SME employees while using ERP systems?*

7.1 Pattern Analysis

The first step of pattern analysis, linking information needs to the usage patterns, is realized by using the content information. This information has been scraped in the data gathering & preprocessing phase. This data consists of the titles of the different aspx'es in the ERP system. These titles give a global description of what information is on the different pages. This content information gives us a global idea of what information a user is looking for on the different pages, and thus what information needs the user has when using the ERP system. Part of the aspx'es and their titles is shown in figure 11 in the content data section of chapter 4.

In order to get an understanding of how the users' information needs can be converted into dashboard components the BI dashboards in multiple ERP systems have been studied. The ERP systems that have been studied for dashboard components are listed in table 11 below. This table also shows what subscription type(s) the dashboards in the system provide information for. The systems taken into account are either free, or could be informed for free by for instance using a free trial period.

Table 11. ERP systems that provided inspiration for dashboard components

Name	Manufacturing	Wholesale	Professional services
Beegua (dashboarding app)			X
Exact Online	X	X	X
Odoo	X	X	X
Qlikview for Exact Globe	X	X	X
Speedbooks (dashboarding-app)	X	X	X
Visionplanner (dashboarding app)	X	X	X

After linking dashboard components to the submenu items based on these two sources of information a number of experts within the case study company have been interviewed. These interviews have been conducted in order to evaluate if the dashboard components linked to the subcategories are relevant to the ERP system of the case study company. Based on these interviews, it has been decided to not include components on assets, campaigns, cost centres, cost units and work centres. After these interviews, the list of dashboard components has been finalized. The final list of components per submenu item is in table 12 below.

Table 12. The final list of possible dashboard components

Submenu item	Component
Accounts	Outstanding items Receivable (Aging)
Bank	Current balance on bank accounts
Contact Persons	List of contact persons (sorted on most recently contacted)
Currency	Exchange rates
Employees	Man/Woman ratio
	Birthdays
Entries	# of unprocessed entries
Entry	Hours per project
	Hours per week
	Billability
G/L Accounts	Revenue, Costs, and Result by Year
Inventory	Total inventory value
Invoices	# of invoice proposals
Items	Top items by revenue
Payroll	Contract types
	FTE Distribution
	Distribution Employees per division
Planning	(See Sales orders)
Projects	# of projects per manager
	# of projects this year/last year
	Project revenue/ this year /last year
Purchase orders	Number of purchase invoices current year
	Total amount in euros purchase invoices current year
	Distribution per GL account
	Distribution per supplier
Quotations	(See Sales orders)
Returns	# of returns
Sales orders	# of Open sales orders
	Amount of open sales orders
Subscriptions	Active subscriptions

As stated above this selection of components per submenu item has been made based on the title elements of the aspx'es belonging to these subcategories. In combination with the knowledge about existing BI dashboards. An example of this is explained below.

Under the submenu item ‘Entry’, fall three aspx’es; "ProDayTransactions.aspx", "ProWeekTransactions.aspx" and “ProTimeCostApproval.aspx”. From the table with aspx’es and title elements, the following three entries are retrieved:

Table 13. Aspx’es and their title, in the submenu item ‘Entry’

Aspx	Title element
ProWeekTransactions.aspx	Weekly entry
ProDayTransactions.aspx	Daily entry
ProTimeCostApproval.aspx	Approval

From this information alone, it can be concluded that the user is interested in daily & weekly entries, and approvals. Combining this information with domain knowledge enriches this conclusion with the fact that these entries have to do with time entries and approvals. Looking into other BI dashboards that have components that are related to time entries it is concluded that a component can be created showing the user entered hours per day or week. Another component often shown together with entered hours is billability. The billability also comes forward in the evaluation with domain experts. Based on the combination of these information sources, the three components shown in table 12 above are shown when a usage pattern contains the submenu item ‘entries’.

7.2 Expanding the model

With results of the pattern analysis stage of web usage mining performed in this chapter, some steps are added to the PDD of our model as presented in figure 23. These steps are specific for the generation of dashboards, based on the usage patterns discovered with the model as described in chapter 6. These extra steps are shown in the PDD in figure 24 below, and are be explained in the text in this section.

To the data gathering activity a step is added that is concerned with gathering content data, in this case this results in the ERP system’s page titles. The data pre-processing activity still has the same steps, but the data cleaning step now also concerns cleaning the content data

A completely new activity called ‘Link information needs’ is added to the PDD. This activity consists of four steps related to linking information needs. First potential information needs are linked to the different submenu items of the ERP system. In order to do this the ERP system’s page titles scraped in the data gathering activity are linked to the pages, by doing this an educated guess on what information is on the pages can be made. The next step is evaluating the potential information needs linked to the different pages of the ERP system with domain experts working at the ERP vendor. After that, potential dashboard components can be linked to the evaluated information needs. These components need to visualize the information that users need in order to fulfill their information need. This step also needs to be evaluated with domain experts working at the ERP vendor. This activity eventually leads to one or more dashboard components linked to each different submenu item

The ‘Assign Usage Pattern’ activity is still the same as in the PDD shown in figure 23. The only difference is that now the resulting usage pattern expresses dashboard components that can fulfill the user’s information needs. Based on this usage pattern and the related dashboard components a user specific dashboard can be generated in the ‘generate dashboard’ activity.

This method has been evaluated with 15 possible end users, using a prototype of the dashboard generator. The prototype is introduced in the next chapter. The evaluation is explained in chapter 9, and the results of the evaluation are discussed in chapter 10.

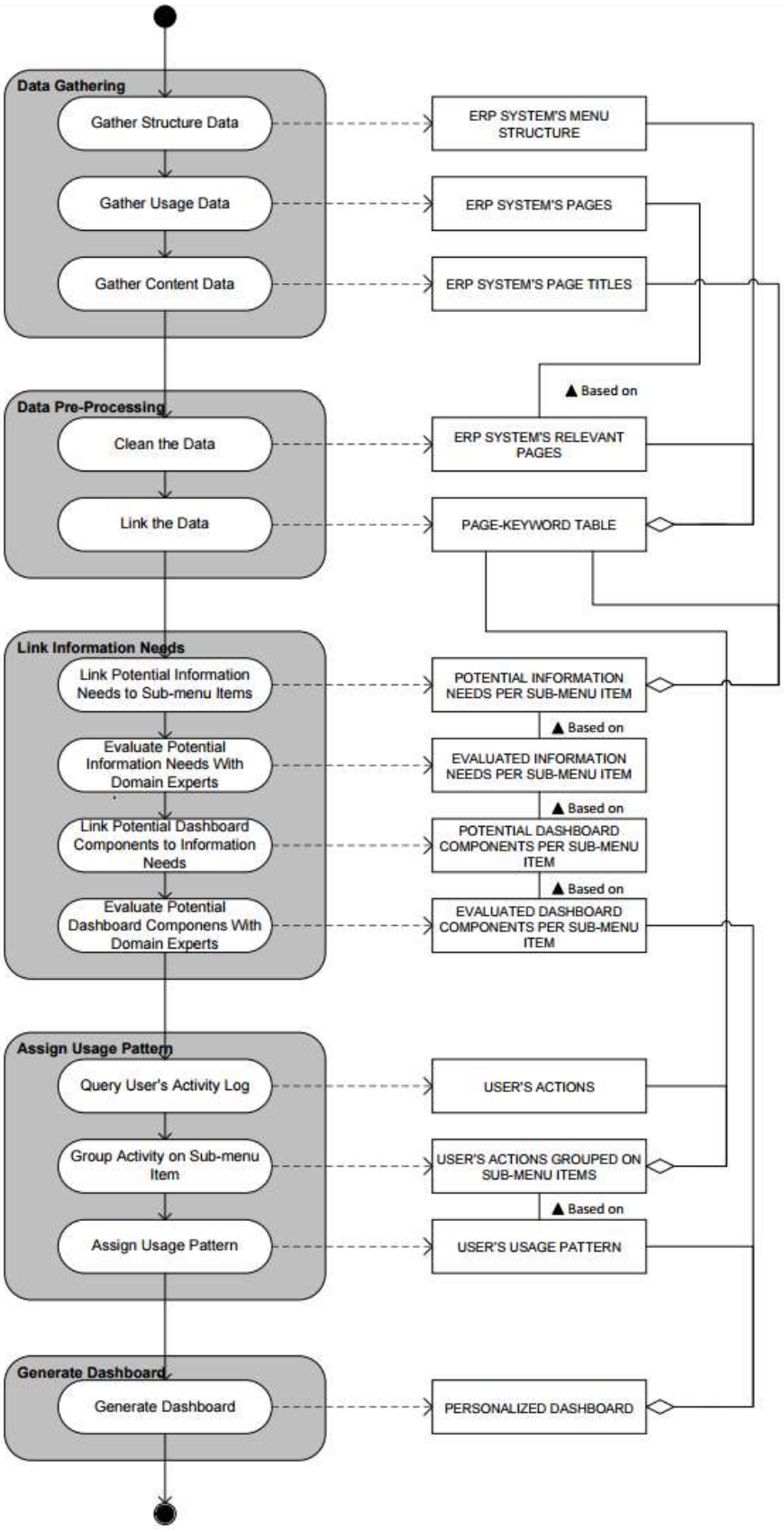

Figure 24. PDD of the dashboard generation method

7.3 Chapter Conclusion

The information needs of different users are hypothetically expressed by the information that is on the pages they visit. In this research the information on the different pages of the ERP system has been gathered by scraping the title elements of these pages. The information on these pages has been linked to dashboard components inspired by existing ERP dashboards, and these components have been evaluated with experts in the field. Linking these dashboard components to the usage data of the ERP system answers the third sub research question: ”How do the usage patterns express different information needs of SME employees while using ERP systems?” Each usage pattern expresses different information needs of users of the ERP system based on their ERP system usage. The last step in the research is now to evaluate the model, and the components linked to the different usage patterns in the model. This last part is completed by employing the prototype of a dashboard generator, as explained in chapter 8, in the case study that is described in chapter 9.

8. Prototype

The last step before the model can be evaluated is creating the prototype of a dashboard generator. This prototype employs the model in order to generate personalized dashboards to users of the ERP system developed by the case study company. The prototype has been developed making use of PHP and Microsoft Power BI. The decision to visualize the dashboards with Microsoft Power BI is made because this tool enables for the fast prototyping of dashboards. The prototype is designed to be a product that when refined is release ready for the case study company.

The first steps in the prototype are concerned with authorization, and the extract, transform, and load (ETL) process. Our case study company provides functionality that enables developers to integrate their application with the SaaS ERP Product. This functionality comes in the form of a public API. Each API request is authenticated using OAuth2.0. Users of the SaaS ERP product can give an app permission to access their ERP data. Figure 24 below show the authentication screen a user is prompted with after logging in to the dashboard application.

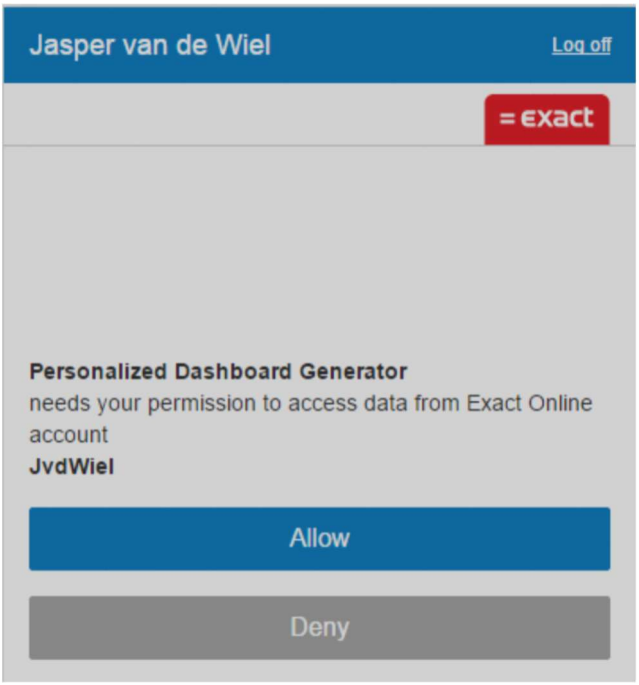

Figure 25. The apps authentication screen

Once the user logs in to the application and allows the application permission to access the users' data, the application has access to the users' User ID. The user ID is checked against a database containing the usage patterns of all EOL users based on their EOL usage of the last 30 days. As explained in chapter 6 the usage pattern is expressed in the form of a string of 1's and 0's based on what subcategories the pages the user has viewed belong to. The usage pattern of the user determines what dashboard components are shown on the dashboard and thus what data is extracted from EOL. As explained in chapter 7 the subcategories have been linked to one or more dashboard components. Each '1' or '0' in the usage pattern stands for a submenu item visited by the user, and a submenu item represents one or more dashboard components. A '1' means this component will be in the dashboard a '0' means it will not be in the dashboard. The data extraction is completed by performing API calls on the ERP system account of the user. The extracted data is stored in a SQL database.

The user is then prompted with a dashboard containing the components based on their usage pattern. Below four screenshots of dashboards are shown. In table 14, their underlying usage pattern as defined by the model explained in chapter 6, are shown. These dashboards are dashboards generated for participants of our case study (Explained in the next chapter). For reasons of confidentiality, the user data is replaced with demo data. The dashboards are in Dutch, because the case study participants have Dutch as their native language.

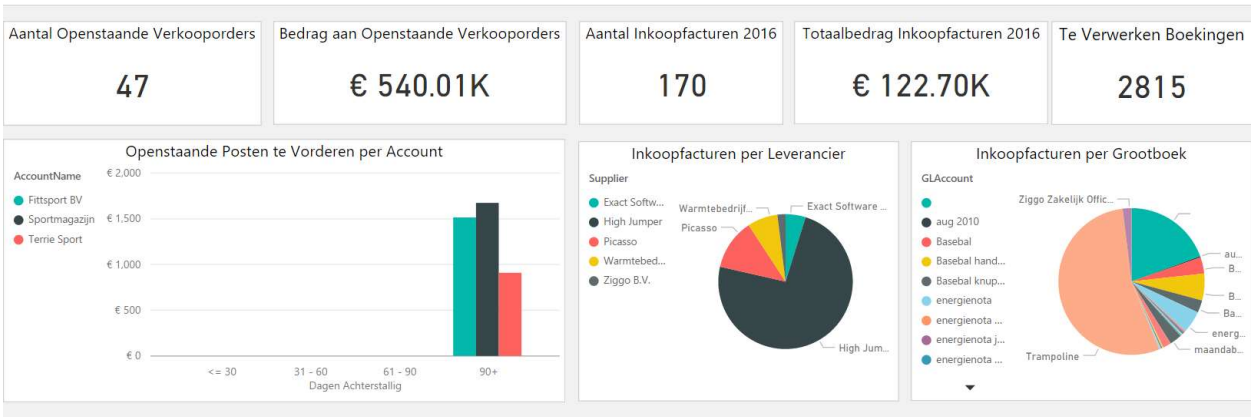

Figure 26. Dashboard for the first usage pattern shown in table 14

Table 12 shows how the usage patterns of four case study participants has been determined. The table names users by a figure number. The dashboards related to these usage patterns are in the figures 25 to 29 below (The data represented in the dashboards has been replaced with dummy data). The usage data of these users has been extracted from the activity log. Combined with the structure data of the ERP system the pages the user visited can be linked to a submenu item. If a user visited five or more pages related to a submenu item, a 1 is linked to this submenu item. If a user visited less than 5 pages related to a submenu item, this submenu item gets a 0 linked to it.

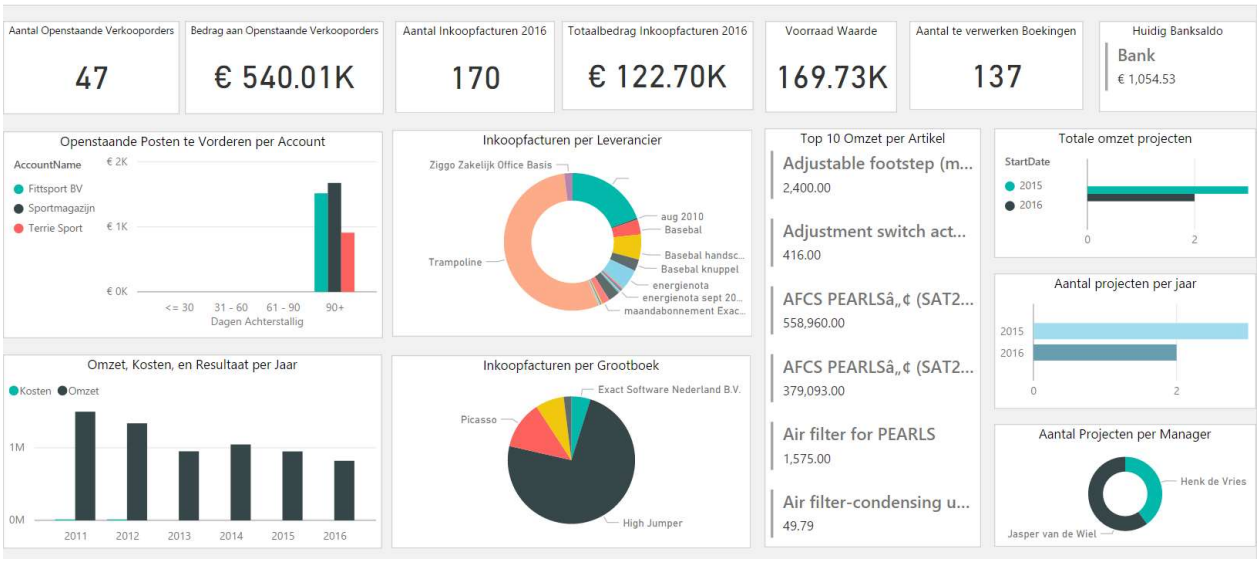

Figure 27. Dashboard for the second usage pattern shown in table 14

Table 14 shows that the user of for instance the dashboard in figure 26 has a usage pattern that is: “10000100000000010010”. The prototype can compare this string to the dashboard components. It will then show the components that are related to the submenu items that have a one linked to them, in the usage pattern, on the dashboard.

Table 14. Usage patterns the dashboards in respectively figure 26, 27, 28 and 29 are based on.

Submenu Item	Figure 26	Figure 27	Figure 28	Figure 29
Accounts	1	1	1	1
Contact Persons	0	0	0	0
Quotations	0	0	0	0
Bank	0	1	1	0
Currency	0	0	0	0
Entries	1	1	1	0
G/L Accounts	0	1	1	0
Employees	0	0	0	0
Payroll	0	0	0	0
Entry	0	0	0	0
Invoices	0	0	1	0
Projects	0	1	0	0
Planning	0	1	0	0
Inventory	0	1	1	0
Purchase Orders	1	1	0	1
Returns	0	0	0	1
Items	0	1	1	1
Sales Orders	1	1	1	0
Subscriptions	0	0	0	0

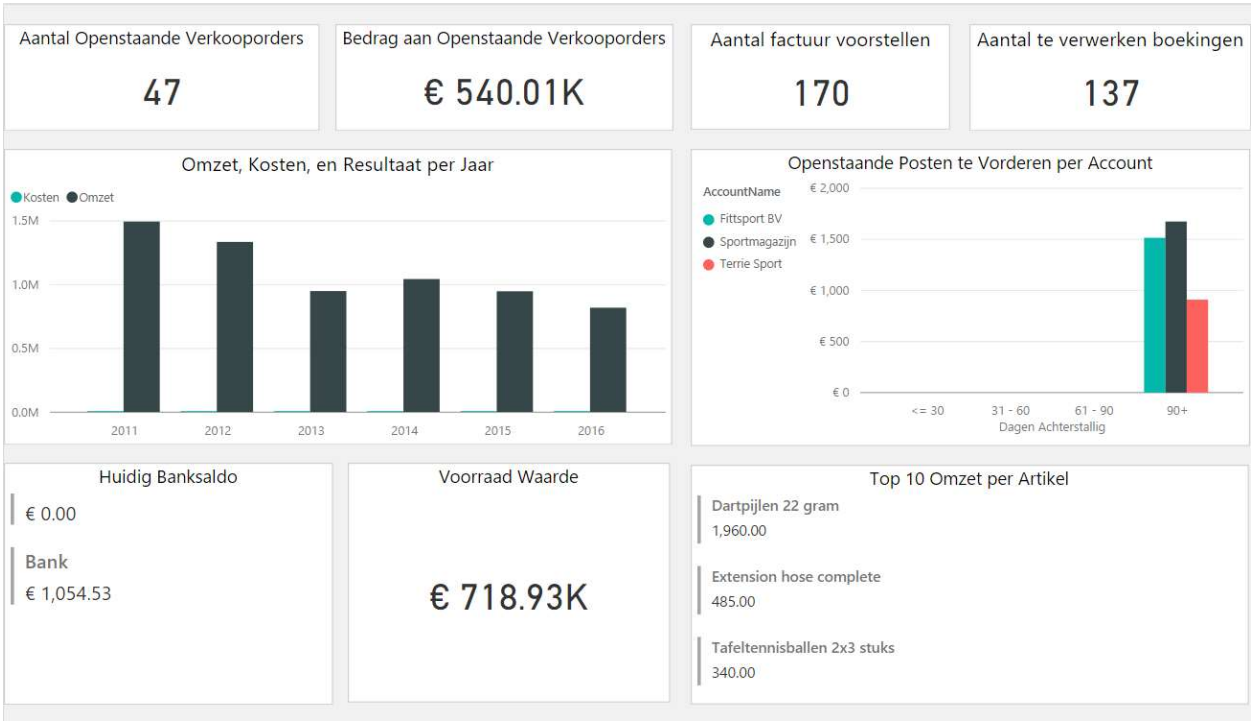

Figure 28. Dashboard for the third usage pattern shown in table 14.

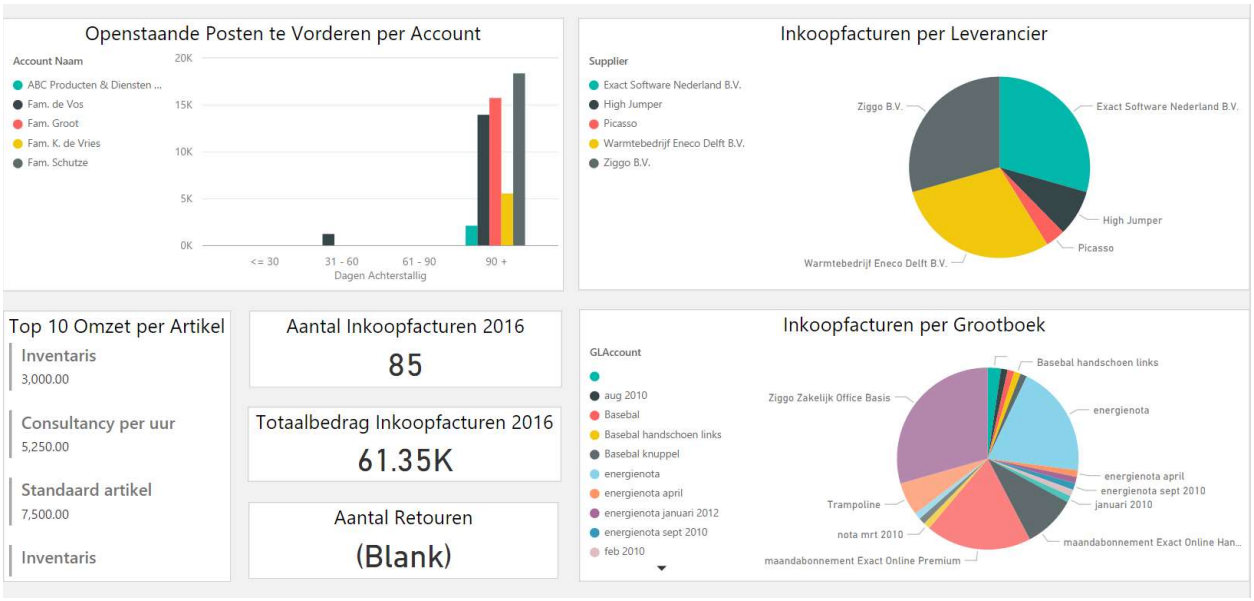

Figure 29. Dashboard for the fourth usage pattern shown in table 14.

9. Evaluation

For the evaluation of the model, a case study has been conducted. The case study company is Exact Software Nederland B.V. (From now on “Exact”). Exact is a business software provider with its headquarters in Delft, The Netherlands. Exact has around 1550 employees based in 15 countries. They serve over 250.000 businesses in 110 countries with their business software. Exact provides both on premise and SaaS products. Since this research is focused on SaaS, the case study is focused on exacts’ SaaS ERP solution called “Exact Online” (Later referred to as EOL). Exact online is a SaaS Multi-Tenant ERP product targeted at SMEs and accountants.

9.1 Case Study Protocol

In the case study, two exercises are employed. The first exercise makes use of a paper prototype, and the second exercise makes use of the developed dashboard generator prototype.

In exercise 1, the case study participant is provided with the 28 components a dashboard can consist of. These components are provided to the participant printed out on paper, containing demo data. The participant is asked to pick their top N preferred components they would like to see in their ideal personal dashboard. The number N is determined based on how many components the model suggests are in the ideal dashboard of that specific user.

After this exercise, the participant is shown the dashboard generated by the dashboard generator based on their user type. This dashboard contains the real user data. Now, the differences between the participants’ picks and the dashboard generated by the prototype are observed. If there are any differences these are discussed. Does the participant still like his/her own selection better than the generators choice? Or did the participant change his/her mind now he/she sees the components with real company data. What is the reason the user does not like the components suggested by the dashboard generator?

After these two exercises, the participant is asked if he/she would have selected more or fewer components than the number they have been asked to select. If this number is lower, the participants are asked to discard the components they would not have picked without the mandatory. Moreover, if this number is higher, the user is asked to name the components they would have liked to include as well. And finally the participant is asked if they miss any components in the list of 28 components they can choose from.

9.2 Case Study Participants

In the case study the prototype is evaluated with 15 participants. All participants are users of the same SaaS ERP system. The number of case study participants beforehand was not predetermined. After twelve evaluation sessions the theoretical saturation, as described by Glaser and Strauss (1967) was achieved. This means the point is reached at which no new information is added to the findings. However, at this point five manufacturing users, five production users, and only two professional services users participated in the case study. Therefore, the decision has been made to speak to three more professional services users in order to have an even distribution between the three packages.

10. Results

In this chapter, the findings of the case study have been documented. Fifteen evaluation sessions have been conducted with customers of the case study company. The case study results answer sub research question 4:

> ***SRQ4**: To what extent does a dashboard, generated based on a model of different usage patterns, satisfy the information needs of individual SME employees?*

The findings are documented per package first, next in the sub-conclusion of this chapter the results of all participants together are summarized (The results of the separate cases can be found in appendix C). Three packages have been studied in this research, and for each package, five case study subjects have conducted the case study assignments. Because of confidentiality, their subscription type, combined with a number, is used in order to distinguish the case study subjects. Per package, the percentage of correct suggestions of components, and subcategories, is given in a table. The distinction between components and subcategories is made because some submenu items are linked to more than one component. This means, if the submenu item prescribes two components for a subcomponent the percentage of correct components is lower if a user only selects one of those two components. Whereas, the percentage of correct suggestions of subcategories is not influenced by this. This distinction enables us to test two phases of our research separately, namely the pattern discovery phase and the pattern analysis phase. The percentage of correct suggestions of subcategories tests the model delivered after pattern discovery. The percentage of correct suggestions of components tests the information needs linked to these subcategories in the pattern analysis phase. Because these percentages alone are not to answer the research question, the case study participants' opinions are taken into account. For this reason, some statements made during the evaluation sessions are included in the text below. Finally, in the conclusion of this chapter sub research question 4 is answered.

In appendix C, each case study evaluation is discussed separately. For each subject a description of what the user uses exact for, if the user wanted to pick more or fewer components than prescribed by the model, and if there are any components they would like to see on a dashboard that are not in the 28 suggested components, are given. Finally, interesting remarks following from the discussion on the differences between the users' picks and the suggestion by the model are described.

10.1 Case Study Results per Package

10.1.1 Manufacturing

Table 15. Case study results of the five manufacturing users

User #	% of correct components	% of correct subcategories
1	80.00	83.33
2	25.00	100.00
3	83.33	80.00
4	86.67	80.00
5	80.00	88.89
Average	71.00	86.44

The users with a manufacturing subscription, in general have a company that produces goods. This can for instance be hospital beds, or cutting dies. As can be seen in table 15 above, the average correct component suggestion for the five case study

participants is 71.00%. The percentage of correct subcategories is higher, at 86.44%. User number 2 causes the substantially lower percentage of correct component suggestion.

"This dashboard is not perfect but it's a good start" – Manufacturing user 3

10.1.2 Wholesale

Table 16. Case study results of the five wholesale users

User #	% of correct components	% of correct subcategories
1	81.82	71.43
2	80.00	90.91
3	66.67	75.00
4	87.50	85.71
5	88.89	87.50
Average	80.98	82.11

The users with a wholesale subscription are in general purchasing, selling and distributing products as an intermediary. This means most wholesale users are working at a business-to-business company. Of the five participants, this holds true for three users. The two others are respectively a company that sells services, and a company with a web shop.

"This dashboard is more relevant to me than the standard dashboards" – Wholesale user 2

As can be seen in table 16 above, the percentage of correct component suggestion is 80.98 %, and the percentage of correct submenu item suggestion is only slightly higher at 82.11%.

"We use the standard dashboards a lot, but a personalized dashboard like this one would be appreciated" - Wholesale user 4

10.1.3 Professional Services

Table 17. Case study results of the five professional services users

User #	% of correct components	% of correct subcategories
1	77.78	83.33
2	76.92	100.00
3	84.62	87.50
4	85.71	88.89
5	83.33	100.00
Average	81.67	91.94

The users of the professional services packages are not as easily generalizable as the previous two packages. Users can have a business in a large range of disciplines. Users can for instance have an advertising agency, a consultancy firm or can be an IT services provider. One of our case study participants runs an advertising agency, and another one runs a design agency.

"I already have these components in my custom dashboard" – PSA User 1

From table 17 above. We see again that with an 81.67% correct component suggestion and a 91.94% correct subcategory suggestion that the correct subcategory suggestion is higher than the correct component suggestion.

10.2 Chapter Conclusion

In order to answer the fourth research question "To what extent does a dashboard, generated based on usage patterns, satisfy the information needs of individual SME employees?" the case study results of the 3 packages can be summarized into two average percentages. These percentages are shown below.

Table 18. Summarized case study results

Package	% of correct components	% of correct subcategories
Manufacturing	71.00	86.44
Wholesale	80.98	82.11
PSA	81.67	91.94
Average	77.88	86.83

As can be seen the average correct component suggestion is at 77.88%, and the average correct submenu item suggestion is at 86.83%. This suggests that the usage patterns are more reliable than the information needs/components linked to these usage patterns. As to answer the research question, the answer can be expressed in the form of these average percentages. More textual conclusions can be deducted from these percentages. Because the results are not 100% accurate, it can be argued that a dashboard, generated based on usage patterns, does not satisfy the information needs of users. An argument for this would be that on average around 1/4th of the components is not suggested correctly. On the other hand, it can be argued that the results are satisfactory because 3/4th of the components is suggested correctly. These opposing statements show that these conclusions are subject to opinion. In order to decide which of the two arguments is most likely true, the opinion of the case study participants can be taken into account.

From the general opinion during the evaluation sessions it can be concluded that the dashboard generator based on usage patterns satisfies the users' information needs more than the standard dashboards they are provided with at the moment. Ten out of the 13 case study participants expressed a positive opinion about the generated dashboard. In general, the overview delivered by our prototype provided the participants with better information, or with equal information but with less noise and without having to change pages, or said that the dashboard was similar to the dashboard they composed themselves.

One of the case study participants mentioned that creating a custom dashboard, as already offered by EOL, has been on his to do list for a while. However, because of the effort this takes when starting from scratch, he has not yet done this. He mentions that if the dashboard generated by the prototype would show up in this section as a start, it would be less work, and he would be more inclined to customize the dashboard to fit his needs. Other case study participants have made similar statements.

Another opinion that has been voiced is that the dashboard shows the same components as the custom dashboard a participant has already created for himself in EOL. This means the generated dashboard does not provide any benefit for this particular participant. However, if this dashboard had shown when he started creating his custom dashboard it would have costed him less time.

If the case study participants' opinions are considered, the conclusion is that the dashboards are satisfactory. In general, the case study participants are happy with more than 3/4th of the components on the suggested dashboard, and the statements made during the evaluation sessions endorse this.

11. Discussion & Future Work

11.1 Discussion

In this chapter, the limitations faced in this research are discussed. First, the internal validity, external validity, and reliability are discussed. Next, other limitations not regarding these three concepts are discussed.

To ensure the internal validity of this research, some steps are taken to limit the chance of errors in the data. All steps for the model development have used the exact same dataset. All evaluation sessions have been conducted using the same interview protocol, and by using the same prototype, components, and dataset.

External validity is related to the ability to generalize this research to subjects other than the case study used in this research. The fact that this research has been conducted using only the data of a single case study company, means some company specific decisions have been made. For instance, focusing on wholesale, production and professional services customers. However, the main contribution of this research is a method that enables a company to specify types of users in their customer base, and generating dashboards based on these user types. This method assumes that a company has an activity log in their product that stores separate user actions. Moreover, the method assumes that there is extra information available to enhance this activity log. Given the fact that almost all ERP systems meet these requirements the method can be generalized to other ERP software companies.

Reliability is related to the ability to repeat this research with the same results. This means other researchers must be able to replicate the steps taken in this research. And by doing so arrive at the same conclusions. For the first part of the research, this is trivial, since the steps from data gathering, to data pre-processing, to pattern discovery, to pattern analysis have been explained in detail. And since real company data has been used, and software that is available for free, in this research the research can be replicated identically. The same goes for the development of the prototype. This leaves us with the question if the opinion of 15 case study participants represents the opinion of all customers. In our opinion 15 case study participants give a good representation of the customer base, since the results from each separate customer interview are similar. This satisfies the theoretical saturation criterion as explained in chapter 9.

After defending the steps taken in this research, we feel the need to discuss some limitations that can be improved in future research. For starters, the activity log at our case study company is not the most detailed activity log possible. The user actions are expressed by aspx requests. This means it is known what aspx a user requests. However, this does not express what the user is doing exactly on this aspx. Certain aspx'es have more than one functionality depending on the parameters given with it in the request. Another limitation in the activity log is that a user can conduct multiple actions on a single aspx without this showing up in the activity log. In future research a case study company with a more detailed activity log could improve the research.

Another limitation on the data at our case study company is the data available for enhancing the activity log. An example of this is the information about company size. This data has been studied in order to find possibilities to enhance the dataset. Unfortunately, it turned out the data about company size was not available for the majority of the customer companies, and if it was available it often turned out not to be reliable. If this data had been available, company size could have played a role in the selection of dashboard components. For instance, not picking the "Projects per Manager" component for companies with 1-5 employees.

11.2 Future work

The results of this research show that a model based on usage and structure data of an ERP system can provide a basis for personalized dashboards. These results provide multiple insights for further research.

The first insight, that follows directly from the difference between the percentage of correct component suggestions and correct subcategory suggestions, is that more research should be performed on the linking of dashboard components to usage patterns. The percentages show that in the prototype subcategory suggestion is better than component suggestion. This is partially explainable by the fact that parameters are not logged in the dataset used for this research, with a more detailed activity log there would be more information available in order to base the component selection on. Another idea for future research is improving the current method and prototype to a point where the successful suggestion percentage reaches 100%. This can be achieved by completing more cycles in the design science process.

Further research could also be performed by not only using the proposed method for the generation of personalized dashboards. With this in mind model-driven development is a field of research that can benefit from the application of usage mining. The method could for instance be extended with the goal of tailoring other components of a SaaS ERP system, or even tailoring the whole system based on usage patterns. Another way of extending the model is not extending the goal of the method, but extending the method to other products. In this research the data of only one SaaS ERP system has been used for the analysis. It is interesting to study how the method performs in for instance another SaaS ERP system. Or maybe even another kind of product that also employs dashboards.

Another way to extend this research, is applying other data mining techniques. In this research a number of techniques have been used in order to discover different usage patterns. The techniques of process mining and FCA in this research have been selected based on their perceived usefulness in dashboard personalization. However, a number of techniques have not been looked into in much detail. For instance, for sequential mining only process mining has been studied in detail. Pachidi et al. (2014) use markov chains in order to discover interesting usage paths in an ERP system.

12. Conclusion

In the first chapter of this research, the main research question has been formulated based on the problem statement. The goal of this research is answering this question. In order to do so, this research has focused on using web usage mining in order to provide user specific BI dashboards to ERP system users. Web usage mining has been used to find different usage patterns in an online SaaS ERP system. Information needs have been linked to these usage patterns, and based on these patterns and information needs, dashboards have been generated that try to satisfy these information needs. The main research question formulated from the problem statement is the following:

> ***RQ:** How can a model of user information needs be developed from ERP usage statistics in order to generate user specific BI dashboards?*

The answer to this research question, and thus the conclusion of this research is: A model of user information needs can be developed from ERP usage statistics by performing web usage mining, applying Formal Concept Analysis as a clustering technique, on a dataset that consists of usage data and structure data. This research also shows that the model can be applied in order to generate user specific BI dashboards. The steps that need to be followed in order to develop a model of usage patterns, and applying this model in order to generate user specific BI dashboard have been expressed in the form of a PDD, the PDD can be found in figure 23 in chapter 7 of this document.

This conclusion follows from the answers to the four sub research questions that have been formulated in chapter 1. The sub research questions each answer part of the main research question. The answers to the sub research questions have been described in the chapters in which each sub research question has been addressed. Below the answers to each of the four research questions that have contributed to the conclusion of this research are summarized.

How can the different usage patterns of SME employees while using ERP systems be identified?

In order to answer this research question, and thus finding a way to identify unique usage patterns of SME employees using ERP systems, first extensive literature research has been performed. The literature research resulted in the conclusion that web usage mining technology has many intersections with the data that is available at ERP software vendors, and with the goal of our research. In web usage mining literature, multiple techniques are found in order to perform web usage mining. In all techniques, the first two steps are data gathering and data pre-processing. These two steps have been executed first. The next step in web usage mining is pattern discovery. A number of pattern discovery techniques has been studied and tried out. Examples of these techniques are association rule mining, process mining, and clustering. These techniques did not succeed in showing us different usage patterns in the data. Eventually we succeeded in identifying unique usage patterns by using Formal Concept Analysis, which is a clustering technique. We did however not succeed by using FCA on only the usage data. This data was too complex and has a low information density. This is why the usage data is combined with structure data. The structure data consist of the menu structure of the ERP system developed by our case study company.

The structure data enables us to group individual page views together in categories based on the menu structure. Page views can be grouped on main-menu item, or more specific on submenu item. By applying FCA on the page views grouped by main-menu item, clusters of usage patterns can be identified. Applying FCA on the page views grouped by submenu item provides a more

detailed view of different clusters of usage patterns. The finding of these patterns proves that usage patterns can be identified by using FCA. In short the answer to first sub-research question 'How can the different usage patterns of SME employees while using ERP systems be identified?' is: By using web usage mining on usage and structure data, employing Formal Concept Analysis as the pattern discovery technique.

What are the different usage patterns of SME employees while using ERP systems?

In order to express the unique usage patterns of SME employees while using ERP systems, and thereby answering the second sub research question formal concept analysis has been applied on the usage data of customers of our case study company. In order to do this, a dataset containing all user actions over the timespan of a month is combined with the structure data expressed by the menu structure of the ERP system. This led to usage patterns on different abstraction layers. The highest level of abstraction is that of menu items. This abstraction level reveals ~100 different usage patterns. Diving one level deeper to the submenu item abstraction level reveals ~3000 usage patterns. Looking at the lowest level of abstraction would be looking at single page views. This is computationally unfeasible and therefore the usage patterns are based on the submenu item abstraction level.

In order to create a model that can be used by others the findings have are described in the form of a simple recipe. In order to find the usage pattern of a user of an ERP system the following steps can be followed:

- Retrieve the menu structure of the ERP system;
- Gather the different pages of the ERP system;
- Clean this data in order to keep only the relevant page views;
- Link the relevant page views to their submenu items in order to form a page-keyword table;
- Retrieve the usage data for the user over a time span of 30 days;
- Link the usage data to the page-keyword table
- If a submenu item appears in the resulting dataset more than 5 times this submenu item is in the usage pattern of the user;
- The combination of submenu items from the above step expresses the usage pattern of this specific user.

These steps have been visualized in a PDD in order to create a clear overview of the steps and their resulting deliverables. These usage patterns are very general, and not yet usable for dashboard generation. In order to generate dashboards based on these usage patterns, information needs have been linked to the submenu items. This process is discussed in the answer to the next research question below.

How do the usage patterns express different information needs of SME employees while using ERP systems?

The information needs of different users are hypothetically expressed by the information that is on the pages they visit. In this research, the information on the different pages of the ERP system has been gathered by scraping the title elements of these pages. The information on these pages has been linked to dashboard components inspired by existing ERP dashboards, and these components have been evaluated with experts in the field. Linking these dashboard components to the usage data of the ERP system answers the third sub research question:"How do the usage patterns express different information needs of SME employees while using ERP systems?" Each usage pattern expresses different information needs of users of the ERP system based on their ERP system usage. The additional steps performed in order to link information needs to usage patterns have been added to the PDD created in answering the previous sub research question. The expanded PDD can be found in chapter 7, in figure 24. The last step in the research is evaluating the model, and the components linked to the different user types in the model.

To what extent does a dashboard, generated based on usage patterns, satisfy the information needs of individual SME employees?

In order to answer this question a prototype of a dashboard generator, which generates dashboards based on the usage patterns expressed by our model has been developed. With this prototype, a case study with 15 participants has been conducted. These participants are users of the ERP system developed by our case study company. Five participants of each package are studied in this research. On average, the model performed best on component suggestion for the professional services package, with a correct suggestion percentage of 81.66%. In addition, on the suggestion of correct submenu items, the model also performed best for the PSA package with a correct suggestion percentage of 91.94%. The average results per package as shown in table 19 below.

Table 19. Summarized case study results

Package	% of correct components	% of correct subcategories
Manufacturing	71.00	86.44
Wholesale	80.98	82.11
PSA	81.67	91.94
Average	77.88	86.83

To answer the research question, to what extent the dashboards satisfies the information needs of individual SME employees in another way than showing the percentages would not be an objective answer. Therefore, the answer to the research question is: The dashboard generated based on usage patterns on average satisfies 77.88% of the user's information needs. For the answer, the correct component suggestion percentage is selected; because this percentage is more related to the actual dashboard the user is confronted with than the percentage of correct submenu item suggestion.

References

Van Der Aalst, W., Adriansyah, A., De Medeiros, A. K. A., Arcieri, F., Baier, T., Blickle, T., ... & Burattin, A. (2011, August). Process mining manifesto. In *International Conference on Business Process Management* (pp. 169-194). Springer Berlin Heidelberg.

Adams, L., & Courtney, J. F. (2004, January). Achieving relevance in IS research via the DAGS framework. In *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on* (pp. 10-pp). IEEE.

Aldridge, G. (2008). Why 70% of Dashboard Initiatives Fail. (White paper)

Aulbach, S., Grust, T., Jacobs, D., Kemper, A., & Rittinger, J. (2008, June). Multi-tenant databases for software as a service: schema-mapping techniques. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data* (pp. 1195-1206). ACM.

Bass, L. (2007). *Software architecture in practice*. Pearson Education India.

Belohlavek, R. (2011, June). What is a Fuzzy Concept Lattice? II⋆. In Rough Sets, Fuzzy Sets, Data mining and Granular Computing: 13th International Conference, RSFDGrC 2011, Moscow, Russia, June 25-27, 2011, Proceedings (Vol. 6743, p. 19). Springer Science & Business Media.

Bernroider, E., & Koch, S. (2001). ERP selection process in midsize and large organizations. *Business Process Management Journal*, *7*(3), 251-257.

Bezemer, C. P., Zaidman, A., Platzbeecker, B., Hurkmans, T., & Hart, A. (2010a, September). Enabling multi-tenancy: An industrial experience report. In *Software Maintenance (ICSM), 2010 IEEE International Conference on* (pp. 1-8). IEEE.

Bezemer, C. P., & Zaidman, A. (2010b, September). Multi-tenant SaaS applications: maintenance dream or nightmare? In *Proceedings of the Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE)* (pp. 88-92). ACM.

Bezemer, C. (2014). *Performance Optimization of Multi-Tenant Software Systems*. TU Delft, Delft University of Technology.

Buxmann, P., Hess, T., & Lehmann, D. W. I. S. (2008). Software as a Service. W*irtschaftsinformatik*, *50*(6),500-503.

Cheng, B. H., & Atlee, J. M. (2007, May). Research directions in requirements engineering. In *2007 Future of Software Engineering* (pp. 285-303). IEEE Computer Society.

Chong, F., & Carraro, G. (2006). Architecture strategies for catching the long tail. *MSDN Library, Microsoft Corporation*, 9-10.

Chou, D. C., Bindu Tripuramallu, H., & Chou, A. Y. (2005). BI and ERP integration. *Information Management & Computer Security*, *13*(5), 340-349.

Cimiano, P., Hotho, A., Stumme, G., & Tane, J. (2004). Conceptual knowledge processing with formal concept analysis and ontologies. In *Concept Lattices*(pp. 189-207). Springer Berlin Heidelberg.

Cimiano, P., Hotho, A., & Staab, S. (2005). Learning Concept Hierarchies from Text Corpora using Formal Concept Analysis. *J. Artif. Intell. Res.(JAIR)*, *24*, 305-339.

D'souza, Austin; Kabbedijk, Jaap; Seo, DongBack; Jansen, Slinger; and Brinkkemper, Sjaak, "Software-As-A-Service: Implications For Business And Technology In Product Software Companies" (2012). PACIS 2012 Proceedings. Paper 140.

Dai, H., & Mobasher, B. (2003, June). A Road Map to More Effective Web Personalization: Integrating Domain Knowledge with Web Usage Mining. In *International Conference on Internet Computing* (pp. 58-64).

Van Dongen, B. F., de Medeiros, A. K. A., Verbeek, H. M. W., Weijters, A. J. M. M., & Van Der Aalst, W. M. (2005, June). The ProM framework: A new era in process mining tool support. In *International Conference on Application and Theory of Petri Nets* (pp. 444-454). Springer Berlin Heidelberg.

Dubey, A., & Wagle, D. (2007). Delivering software as a service. *The McKinsey Quarterly*, *6*(2007), 2007.

Eckerson, W. (2003). Smart companies in the 21st century: The secrets of creating successful business intelligence solutions. *TDWI Report Series*, *7*.

Eckerson, W. (2010). *Performance dashboards: measuring, monitoring, and managing your business*. John Wiley & Sons.

Eekels, J., & Roozenburg, N. F. (1991). A methodological comparison of the structures of scientific research and engineering design: their similarities and differences. *Design Studies*, *12*(4), 197- 203.

Engebrethson, R. (2012). Comparative Analysis of ERP Emerging Technologies.

Van Everdingen, Y., Van Hillegersberg, J., & Waarts, E. (2000). Enterprise resource planning: ERP adoption by European midsize companies. *Communications of the ACM*, *43*(4), 27-31.

Evelson, B. (2010). The Forrester Wave™: Enterprise Business Intelligence Platforms, Q4 2010. *Intelligence (BI)*, *3*.

Fayyad, U. M., Piatetsky-Shapiro, G., & Smyth, P. (1996, August). Knowledge discovery and data mining: towards a unifying framework. In *KDD* (Vol. 96, pp. 82-88).

Fensel, D. Ontologies: A silver bullet for knowledge management and electronic-commerce (2000). *Berlin: Spring-Verlag*.

Few, S. (2006). *Information dashboard design* (pp. 120-206). O'Reilly.

Formica, A. (2006). Ontology-based concept similarity in formal concept analysis. *Information Sciences*, *176*(18), 2624-2641.

Fui-Hoon Nah, F., Lee-Shang Lau, J., & Kuang, J. (2001). Critical factors for successful implementation of enterprise systems. *Business process management journal*, *7*(3), 285-296.

Frank, U., Heise, D., & Kattenstroth, H. (2009). Use of a domain specific modeling language for realizing versatile dashboards. In *Proceedings of the 9th OOPSLA workshop on domain-specific modeling (DSM)*.

Ganter, B., Stumme, G., & Wille, R. (2002). Formal concept analysis: Methods and applications in computerscience. *TU Dresden, http://www. aifb. uni-karlsruhe. de/WBS/gst/FBA03. shtml*.

Ganter, B., & Wille, R. (2012). *Formal concept analysis: mathematical foundations*. Springer Science & Business Media.

Gile, K. (2006). The Forrester Wave™: BI Reporting And Analysis Platforms, Q1 2006. *Forrester, Fevereiro*.

Glaser, B. G., & Strauss, A. L. (1998). Grounded theory. *Strategien qualitativer Forschung. Bern: Huber*.

Guarino, N. (1998). *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy* (Vol. 46). IOS press.

Guo, C. J., Sun, W., Jiang, Z. B., Huang, Y., Gao, B., & Wang, Z. H. (2011). Study of software as a service support platform for small and medium businesses. In *New Frontiers in Information and Software as Services* (pp. 1-30). Springer Berlin Heidelberg.

Griffin, J. (2007). BI and ERP integration: Five critical questions. *Information Management*, *17*(5), 6.

Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing?. *International journal of human-computer studies*, *43*(5), 907-928.

Haddara, M., & Zach, O. (2011, January). ERP systems in SMEs: A literature review. In *System Sciences (HICSS), 2011 44th Hawaii International Conference on* (pp. 1-10). IEEE.

Hevner, March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, *28*(1), 75–105.

Jansen, S., Houben, G. J., & Brinkkemper, S. (2010). Customization realization in multi-tenant web applications: case studies from the library sector (pp. 445-459). Springer Berlin Heidelberg.

Jones, D. (2009). *The Shorcut Guide to Achieving Business Intelligence in Midsize Companies*.Realtimepublishers.com.

Kabbedijk, J., & Jansen, S. (2011). Variability in multi-tenant environments: architectural design patterns from industry. In *Advances in conceptual modeling. Recent developments and new directions* (pp. 151-160). Springer Berlin Heidelberg.

Kabbedijk, J., Pors, M., Jansen, S., & Brinkkemper, S. (2014). Multi-tenant Architecture Comparison. In *Software Architecture* (pp. 202-209). Springer International Publishing.

Kaplan, M.J. 2005. SaaS Survey Shows New Model Becoming Mainstream. Cutter Consortium Executive Update, 6(22), 1-5.

Kaplan, J. M. (2007). Saas: Friend or foe?. Business Communications Review, 37(6), 48.

Kawamoto, T., & Mathers, B. (2007). Key success factors for a performance dashboard. *Information Management*, *17*(7), 20.

Kumar, K., & van Hillegersberg, J. (2000). Enterprise resource planning: Introduction. *Communications of the ACM*, *43*(4), 22-26.

Kwok, T., T. Nguyen, and L. Lam (2008). "A software as a service with multi-tenancy support for an electronic contract management application". In: Proceedings of the International Conference on Services Computing (SCC). IEEE, pp. 179–186.

Lidwell, W., Holden, K., & Butler, J. (2010). *Universal principles of design, revised and updated: 125 ways to enhance usability, influence perception, increase appeal, make better design decisions, and teach through design*. Rockport Pub.

Liu, B. (2007). Web usage mining. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*, 449-483.

Mäkilä, T., Järvi, A., Rönkkö, M., & Nissilä, J. (2010). How to Define Software-as-a-Service–An Empirical Study of Finnish SaaS Providers. In *Software Business* (pp. 115-124). Springer Berlin Heidelberg.

Makkar, G. D., & Bist, M. (2012). EaaS-ERP as a Service. *Journal of Information and Operations Management*, *3*(1), 141.

Mertz, S. A., Eschinger, C., Eid, T., & Pring, B. (2007). Dataquest insight: SaaS demand set to outpace enterprise application software market growth. *Gartner report*.

Mobasher, B., Cooley, R., & Srivastava, J. (2000). Automatic personalization based on web usage mining. *Communications of the ACM*, *43*(8), 142-151.

Mobasher, B. (2007). Data mining for web personalization. In *The adaptive web* (pp. 90-135). Springer Berlin Heidelberg.

Nitu (2009). Configurability in SaaS (software as a service) applications. In *Proc. 2nd Annual Conference on India Software Engineering Conference* (pp. 19-26).

Nunamaker Jr, J. F., & Chen, M. (1990, January). Systems development in information systems research. In *System Sciences, 1990., Proceedings of the Twenty-Third Annual Hawaii International Conference on* (Vol. 3, pp. 631-640). IEEE.

Nuseibeh, B., & Easterbrook, S. (2000, May). Requirements engineering: a roadmap. In *Proceedings ofthe Conference on the Future of Software Engineering* (pp. 35-46). ACM.

Obitko, M., Snasel, V., Smid, J., & Snasel, V. (2004, September). Ontology Design with Formal Concept Analysis. In *CLA* (Vol. 110).

Öhgren, A., & Sandkuhl, K. (2005, February). Towards a methodology for ontology development in smalland medium-sized enterprises. In *IADIS AC* (pp. 369-376).

Pachidi, S., Spruit, M., & Van De Weerd, I. (2014). Understanding users' behavior with software operation data mining. *Computers in Human Behavior*, *30*, 583-594.

Palpanas, T., Chowdhary, P., Mihaila, G., & Pinel, F. (2007). Integrated model-driven dashboard development. *Information Systems Frontiers*, *9*(2-3), 195-208.

Papazoglou, M. P. (2003, December). Service-oriented computing: Concepts, characteristics and directions.In *Web Information Systems Engineering, 2003. WISE 2003. Proceedings of the Fourth International Conference on* (pp. 3-12). IEEE.

Peffers, K., Tuunanen, T., Gengler, C. E., Rossi, M., Hui, W., Virtanen, V., & Bragge, J. (2006, February). The design science research process: a model for producing and presenting information systems research. In *Proceedings of the first international conference on design science research in information systems and technology (DESRIST 2006)* (pp. 83-106).

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research.*Journal of management information systems*, *24*(3), 45-77.

Priss, U. (2006). Formal concept analysis in information science. *Arist*, *40*(1), 521-543.

van Rees, R. (2003). Clarity in the usage of the terms ontology, taxonomy and classification. *CIB REPORT*, *284*(432), 1-8.

Rossi, M., & Sein, M. K. (2003). Design research workshop: a proactive research approach. *Presentation delivered at IRIS*, *26*, 9-12.

Sharda, R., Aronson, J. E., & King, D. (2008). *Business intelligence: A managerial approach*. Upper Saddle River: Pearson Prentice Hall.

Srivastava, J., Cooley, R., Deshpande, M., & Tan, P. N. (2000). Web usage mining: Discovery and applications of usage patterns from web data. *Acm Sigkdd Explorations Newsletter*, *1*(2), 12-23.

Steinebach, G., Guhathakurta, S., & Hagen, H. (Eds.). (2009). *Visualizing sustainable planning*. Springer.

Stumme, G. (2009). Formal concept analysis. In *Handbook on Ontologies* (pp. 177-199). Springer Berlin Heidelberg.

Takeda, H., Veerkamp, P., & Yoshikawa, H. (1990). Modeling design process. *AI magazine*, *11*(4), 37.

Thompson, O. (2004). Business Intelligence Success, Lessons Learned. Technology.

Tutunea, M. F., & Rus, R. V. (2012). Business intelligence solutions for SME's. *Procedia Economics and Finance*, *3*, 865-870.

Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, methods and applications. *Knowledge engineering review*, *11*(2), 93-136.

Walls, J. G., Widmeyer, G. R., & El Sawy, O. A. (1992). Building an information system design theory for vigilant EIS. *Information systems research*, *3*(1), 36-59.

Wang, Z. H., Guo, C. J., Gao, B., Sun, W., Zhang, Z., & An, W. H. (2008, October). A study and performance evaluation of the multi-tenant data tier design patterns for service oriented computing. In *e-Business Engineering, 2008. ICEBE'08. IEEE International Conference on* (pp. 94-101). IEEE.

Wang, L., Von Laszewski, G., Younge, A., He, X., Kunze, M., Tao, J., & Fu, C. (2010). Cloud computing: a perspective study. *New Generation Computing*, *28*(2), 137-146.

Watson, H. J., Wixom, B. H., Hoffer, J. A., Anderson-Lehman, R., & Reynolds, A. M. (2006). Real-time business intelligence: Best practices at Continental Airlines. *Information Systems Management*, *23*(1), 7.

Watson, H. J., & Wixom, B. H. (2007). The current state of business intelligence. *Computer*, *40*(9), 96-99.

Wieringa, R. (2009, May). Design science as nested problem solving. In Proceedings of the 4th international conference on design science research in information systems and technology (p. 8). ACM.

Wixom, B., & Watson, H. (2012). The BI-based organization. Organizational Applications of Business Intelligence Management: Emerging Trends, IGI Global, Hershey, 193-208.

Xu, L. D. (2011). Enterprise systems: state-of-the-art and future trends. *Industrial Informatics, IEEE Transactions on*, *7*(4), 630-640.

Appendices

Appendix A. Aspx'es, Related (Sub)Menu Items, and Title Elements

Category	Subcategory	Aspx	Title element
Accountancy	General	AccBatchReports.aspx	Management|Batch reports
Accountancy	General	AccCompareFinPeriods.aspx	Compare financial periods
Accountancy	Management	AccCustomerManagement.aspx	Management|Master data
Accountancy	General	AccGLJournalOverview.aspx	G/L Account - Journal
Accountancy	Management	AccLiquidAssets.aspx	Liquidity overview per company
Accountancy	Management	AccProcessManagement.aspx	Financial process management
Accountancy	Management	AccTransManagement.aspx	Management|Transactions
Accountancy	Management	AccVATManagement.aspx	Management: VAT return
Financial	Taxes and returns	AccVATMatching.aspx	VAT-matching
Financial	Assets	AssAssetList.aspx	Overview: Assets
Financial	Assets	AssAssetStatements.aspx	Assets statement
Financial	Assets	AssAssetTransactions.aspx	Transactions: Asset
Financial	Assets	AssBalanceList.aspx	Assets-Checklist
Financial	Assets	AssDeprPlanning.aspx	Deprecation Plan
Financial	Assets	AssHistorySheet.aspx	Assets History sheet
Financial	Assets	AssInvestmentDeduct.aspx	Invesetment - Deduction
Financial	Assets	AssInvestmentList.aspx	Investment - List
Financial	Bank	BOPspPayments.aspx	Payments
Financial	Bank	CflAllocationRules.aspx	Overview: Allocation rules
Financial	Bank	CflBankFiles.aspx	Bank files
Financial	Bank	CflBatchBookingOverview.aspx	Overview: Direct debit batches
Financial	Bank	CflCashflowTransactions.aspx	Overview outgoing cheques
CRM	Accounts	CflCMScenarioActions.aspx	Management: Credit
Financial	Bank	CflDDMandates.aspx	Overview: Mandates
Financial	Bank	CflFinancialStatements.aspx	Overview statements
Financial	Bank	CflFinancialStmtEntry.aspx	Enter Manual statement
Financial	Bank	CflImport.aspx	Import statements
Financial	Bank	CflLinks.aspx	Bank Links
Financial	Bank	CflPayments.aspx	Payments
Financial	Bank	CflPaymentsCreditCard.aspx	Select Creditcard Payments
Financial	Bank	CflPaymentsNew.aspx	Pay by cheque
Financial	Bank	CflRemittances.aspx	Remittance adivce
Financial	Bank	CflStatements.aspx	To be reconciled
CRM	Campaigns	CMPCampaign.aspx	Campagin
CRM	Campaigns	CMPCampaigns.aspx	Overview: Campaigns
CRM	Accounts	CRMACcountNew.aspx	Create: Account
CRM	Accounts	CRMAccounts.aspx	Overview: Accounts
CRM	Accounts	CRMAccountSearch.aspx	Search: Accounts

CRM	Accounts	CRMAccountsimport.aspx	Import accounts
CRM	Accounts	CRMACcountValidation.aspx	Accounts: Validation
CRM	Accounts	CRMAccountValList.aspx	Accounts: Validation
CRM	Contact persons	CrmContact.aspx	Contact
CRM	Accounts	CRMContacts.aspx	Overview: Contacts
CRM	Accounts	CRMContactSearch.aspx	Search: Contacts
Financial	Bank	CrmSepaCustomerBankAccounts.aspx	Debtor: Bank accounts
CRM	Accounts	FinARAPAging.aspx	Ageing analysis: A/R
CRM	Accounts	FinARAPBalance.aspx	Balance list: A/R
CRM	Accounts	FinARAPOutstanding.aspx	Outstanding items: A/R
CRM	Accounts	FINARAPReminders.aspx	Print Reminders
Financial	G/L Accounts	FinBalanceSheet.aspx	Balance sheet / Profit and loss statement
Financial	Bank	FinBankEntities.aspx	Overview|Bank accounts
Financial	Budgets	FinBudgets.aspx	Budgets
Financial	Budgets	FinBudgetScenarios.aspx	Overview: Budget scenarios
Financial	Entries	FinFixedEntries.aspx	Overview: Recurring entries
Financial	G/L Accounts	FinGLAccount.aspx	Overview: G/L Accounts
Financial	Taxes and returns	FinIntrastatReturns.aspx	Overview: INTRASTAT returns
Sales	Sales Orders	FinInvoices.aspx	Print invoices
Financial	Entries	FinJournalSelect.aspx	Entry - New
Financial	Entries	FinMissingEntryNumbers.aspx	Overview: Missing entry numbers
Financial	G/L Accounts	FinPeriodDateCheck.aspx	Period-date check
Financial	G/L Accounts	FinProcessResults.aspx	Year end closing: Process profit / loss
Financial	Currency	FinRevaluation.aspx	Currency - Revaluation
Financial	G/L Accounts	FinTransactionSearch.aspx	Search: Financial - Transactions
Financial	Entries	FinUnprocessedEntries.aspx	Overview: Entries
Financial	Taxes and returns	FinVATCheckList.aspx	VAT checklist
Financial	Taxes and returns	FinVatReturns.aspx	VAT returns
Financial	Taxes and returns	FinVATSalesLists.aspx	EC sales list
Financial	Taxes and returns	FinVATTransactions.aspx	VAT overview
CRM	Quotations	GenProcessJobs.aspx	Print Quotations
Accountancy	General	GenTopicDefineTopics.aspx	Define template
Person	Payroll	HRMAbsencePercentages.aspx	Absence percentages: Overview
Person	Absence	HRMAbsenceRegs.aspx	Overview: Leave registrations
Financial	Cost centres	HRMCostCenters.aspx	Overview: Cost centres
Financial	Costs units	HRMCostUnits.aspx	Overview: Cost units
Person	Employees	HRMEmplAnniversaries.aspx	Overview: Anniversaries
Person	Employees	HRMEmplBirthdayList.aspx	Birthday list
Person	Employees	HRMEmplContracts.aspx	Overview: Employment contracts
Person	Employees	HRMEmployees.aspx	Overview: Employees
Person	Absence	HRMIllnessRegs.aspx	Overview: Absence registrations
Person	Employees	HRMPFEmployees.aspx	Overview|Pro forma employees
Person	Payroll	HRMSchedules.aspx	Overview: Schedules

Inventory	Planning	InvSalesForecasts.aspx	Sales forecast
Purchase	Inventory	InvStockCountNew.aspx	Stock count - New
Purchase	Inventory	InvStockCounts.aspx	Overview: Stock counts
Purchase	Inventory	InvStockPositions.aspx	Stock positions
Purchase	Inventory	InvStockValue.aspx	Stock value
Sales	Items	InvTransactions.aspx	Transaction list
Purchase	Warehouse transfers	InvWarehouseTransfer.aspx	Warehouse transfer
Purchase	Warehouse transfers	InvWarehouseTransferDeliveries.aspx	Print warehouse transfer
Purchase	Warehouse transfers	InvWarehouseTransfers.aspx	Overview: Warehouse transfers
Manufacturing	Realization	LicDivisionSettings.aspx	Company settings
Sales	Sales Orders	LogEntries.aspx	Overview: Sales invoices
Sales	Sales Orders	LogFixedEntries.aspx	Overview: Recurring sales invoices
Sales	Items	LogItem.aspx	Item
Purchase	Inventory	LogItemCostPriceChanges.aspx	Cost price changes
Sales	Items	LogItemGroups.aspx	Overview: Item groups
Sales	Items	LogItemLabelsPrint.aspx	Print item labels
Sales	Items	LogItemPrices.aspx	Overview: Item prices
Sales	Items	LogItemPriceUpdate.aspx	Update prices
Sales	Items	LogItems.aspx	Overview: Items
CRM	Marketing Lists	MarMarketingLists.aspx	Overview: Marketing lists
CRM	Marketing Lists	MarMarketingListWizard.aspx	New: Marketing list
Accountancy	General	MenuMCM.aspx	Companies
Manufacturing	Planning	MfgDemandPlanning.aspx	Demand planning
Manufacturing	Realization	MfgIssueMaterials.aspx	Issue materials - Selection
Manufacturing	Realization	MfgIssueReversalHistory.aspx	Issue reversal history
Manufacturing	Realization	MfgIssueReversals.aspx	Issue reversal - New
Manufacturing	Operations	MfgOperation.aspx	Operation
Manufacturing	Operations	MfgOperations.aspx	Overview: Operations
Manufacturing	Realization	MfgOperationsToPurchaseOrders.aspx	Overview: Purchase subcontracts
Manufacturing	Work centres	MfgProductionAreas.aspx	Overview: Production areas
Manufacturing	Planning	MfgShopFloorPlanning.aspx	Shop floor planning
Manufacturing	Planning	MfgShopOrder.aspx	Shop order
Manufacturing	Realization	MfgShopOrderComplete.aspx	Complete: Shop orders
Manufacturing	Realization	MfgShopOrderFinishItem.aspx	Finish shop orders
Manufacturing	Realization	MfgShopOrderFinishItemReturnHistory.aspx	Finish reversal history
Manufacturing	Realization	MfgShopOrderFinishItemReturns.aspx	Finish reversal - New
Manufacturing	Planning	MfgShopOrders.aspx	Overview: Shop orders
Manufacturing	Realization	MfgShopOrderStockReceiptPrint.aspx	Print shop order stock receipts
Manufacturing	Planning	MfgShopPapersPrint.aspx	Print shop paper
Manufacturing	Realization	MfgSubcontractDeliveryPrint.aspx	Print subcontract delivery notes
Manufacturing	Work centres	MfgWorkcenter.aspx	Work centre
Manufacturing	Work centres	MfgWorkcenters.aspx	Overview: Work centres
Person	Payroll	PrlCheckPeriods.aspx	Overview: Check

Person	Payroll	PrlCompGroupCLAS.aspx	Overview|Employment conditions groups
Person	Payroll	PrlComponentsDiv.aspx	Overview: Calculations
Person	Payroll	PrlControlRegister.aspx	Control Register
Person	Payroll	PrlCorrections.aspx	Overview: Correction requests
Person	Payroll	PrlEmplCostRegulation.aspx	Overview: Employment cost regulation
Person	Payroll	PrlEntries.aspx	Variable mutations
Person	Payroll	PrlHistoricalData.aspx	Overview: Historical entries
Person	Employees	prlMassUpdateSelection.aspx	Mass update : Selection
Person	Payroll	PrlPayments.aspx	Overview: payments
Person	Payroll	PrlPaySlips.aspx	Payroll slips
Person	Payroll	PrlProcessesOverview.aspx	Payroll process management
Person	Payroll	PrlRepJournalEntries.aspx	Journal entries
Person	Payroll	PrlReportsOverview.aspx	Overview reports
Person	Payroll	PrlTaxDeclarations.aspx	Overview: Tax declaration
Person	Payroll	PrlTaxReturns.aspx	Payroll taxes declaration
Person	Payroll	PrlVariableMutations.aspx	Variable mutations
Person	Payroll	PrlWagelist.aspx	Wage list
Person	Payroll	PrlYearEndOVerview.aspx	Year process management
Person	Payroll	PrlYearOverview.aspx	Income statements
Person	Payroll	PrlYearPEriods.aspx	Generate: Period-date table
Project	Projects	ProBucketPlanningBoard.aspx	Planning Board
Project	Entry	ProDayTransactions.aspx	Daily entry
Project	Invoices	ProInvoiceSpecification.aspx	Print invoice specification
Project	Projects	ProMyPlanning.aspx	My Planning
Project	Projects	ProProject.aspx	Project
Project	Projects	ProProjects.aspx	Overview: Projects
Project	Projects	ProRepBillability.aspx	Report: Billability
Project	Entry	ProTimeCostApproval.aspx	Approve
Project	Entry	ProWeekTransactions.aspx	Weekly entry
Sales	Subscriptions	PsnGenerateInvoices.aspx	Generate invoices
Sales	Subscriptions	PsnPrintSubscriptions.aspx	Print subscriptions
Sales	Subscriptions	PsnProlongation.aspx	Renew: Subscriptions
Sales	Subscriptions	PsnSubscription.aspx	Create: Subscription
Sales	Subscriptions	PsnSubscriptions.aspx	Overview: Subscriptions
Sales	Subscriptions	PsnSubscriptionSearch.aspx	Search: Subscriptions
Sales	Subscriptions	PsnUpdatePrices.aspx	Update subscription prices
Purchase	Purchase orders	PurBalanceList.aspx	Status overview: Purchase orders
Purchase	Purchase orders	PurEntryPurchOrder.aspx	Purchase order
Purchase	Items	PurPurchAnalysis.aspx	Purchase analysis
Purchase	Purchase invoices	PurPurchInvoices.aspx	Overview: Direct purchase invoices
Purchase	Purchase orders	PurPurchOrders.aspx	Overview: Purchase orders
Purchase	Goods receipts	PurPurchOrderSelect.aspx	Goods receipt - New
Purchase	Purchase orders	PurPurchOrdersPrint.aspx	Print purchase orders

Purchase	Goods Receipts	PurPurchOrdersToBeInvoiced.aspx	To be billed
Purchase	Returns	PurReceiptReturns.aspx	Print purchase return
Purchase	Goods receipts	PurReceipts.aspx	Print receipt slip
Purchase	Returns	PurReceiptSelect.aspx	Purchase return - New
CRM	Quotations	QuoEntryQuotation.aspx	Quotation
CRM	Quotations	QuoPrintQuotations.aspx	Print quotations
CRM	Quotations	QuoQuotations.aspx	Overview: Quotations
CRM	Quotations	QuoQuotationsApprove.aspx	Approve
CRM	Accounts	RepAccountCard.aspx	Transactions: Debtors
Financial	Costs units	RepAnalytical.aspx	Analytical values
Financial	Entries	RepEntryReport.aspx	Entry report
Financial	G/L Accounts	RepGLCardReport.aspx	General ledger transactions
Person	Payroll	RepPayrollPayments.aspx	Payment advice list
Person	Payroll	RepPrlAnnualReport.aspx	Year overview employee
Person	Payroll	RepPrlJrnlEntries.aspx	Journal entries
Person	Employees	RepPrlNoPrvtCar.aspx	Statement of no private use car
Person	Employees	RepPrlTaxStatement.aspx	Statement payroll taxes
Person	Payroll	RepPrlWagelist.aspx	Wage lists
CRM	Leads	SFALeadLifeTime.aspx	Lead Lifetime
CRM	Leads	SFALeads.aspx	History Lead
CRM	Opportunities	SFAOpportunities.aspx	Overview: Opportunities
CRM	Opportunities	SFAOpportunity.aspx	Opportunity
Project	Invoices	SlsInvoicePlanningAccounts.aspx	Proposals
Sales	Sales Orders	SlsMergeSalesInvoices.aspx	Merge invoices
Sales	Sales Orders	SlsPickingListPrintOnly.aspx	Pickinglist
Sales	Sales Orders	SlsPickingLists.aspx	PickingLists
Sales	Sales Orders	SlsRetToBeCredited.aspx	To be credited
Sales	Sales Orders	SlsSalesMargin.aspx	Sales margin (Summary)
Sales	Sales Orders	SlsSalesOrderEntry.aspx	Create: Sales order
Sales	Sales Orders	SlsSalesOrderPicklist.aspx	Sales order: Picking List
Sales	Sales Orders	SlsSalesOrders.aspx	Overview: Sales orders
Sales	Sales Orders	SlsSalesOrdersApprove.aspx	Approve
Purchase	Purchase Orders	SlsSalesOrderSelectToCreatePurchOrders.aspx	Back to back order
Sales	Sales Orders	SlsSalesOrdersPrint.aspx	Print sales orders
Sales	Sales Orders	SlsSalesordersTobeInvoicedAdv.aspx	To be invoiced
Sales	Sales Orders	SlsShipmentReturns.aspx	Sales return - New
Sales	Sales Orders	SlsShipmentReturnsPrint.aspx	Sales return history
Sales	Sales Orders	SlsShipments.aspx	Goods delivery - New
Sales	Sales Orders	SlsShipmentsPrint.aspx	Delivery history
Sales	Sales Orders	SlsStatusList.aspx	Status overview: Sales orders
Financial	Currency	SysCurrencies.aspx	Overview: Exchange rates
Person	Employees	WizNewEmployee.aspx	New: Employee

Appendix B. Deleted Aspx'es

Aspx	Reason
AssCDioxide.aspx	Country Specific
AssHistorySheet.aspx	Country Specific
ConMutations.aspx	Functionality for Resellers Only
CRMCreditChecker.aspx	Credit Check
CrmSocialBusinessTweets.aspx	Too General
FinBalanceList.aspx	Too General
FinBalanceReport.aspx	Country Specific
FinBWA.aspx	Country Specific
FinCentralizeGLAccount.aspx	Country Specific
FinDebtorStatements.aspx	Country Specific
FinEmpCostRegulations.aspx	Country Specific
FinEUSalesListBECorr.aspx	Country Specific
FinExpenseNonDed.aspx	Country Specific
FinPrivateUse.aspx	Country Specific
FinSupplierPaymentReporting.aspx	Report
FinTransactions.aspx	Too General
FinWithholdingModel111.aspx	Country Specific
FinWithholdingModel115.aspx	Country Specific
InvItemMRPOverview.aspx	Report
InvStockAgingAnalysis.aspx	Report
InvStockLocations.aspx	Report
InvWhereUsed.aspx	Report
IppSalesCreditMemos.aspx	Too specific for dashboard
IppSalesInvoices.aspx	Too specific for dashboard
LicHostingCustomers.aspx	Functionality for Resellers Only
LicResellerLinks.aspx	Functionality for Resellers Only
MenuPortal.aspx	Expresses no specific information
MfgBackflushOverview.aspx	Report
MfgIndentedBillOfMaterials.aspx	Report
MfgProductionAnalysis.aspx	Report
MfgShopOrderCost.aspx	Report
MfgShopOrderResults.aspx	Report
MfgShopOrderStatus.aspx	Report
PrlPayrollExport.aspx	Expresses no specific information
ProCustomerTrans.aspx	Report
ProRepCapacity.aspx	Report
ProRepCapacityForecast.aspx	Report
ProRepCosts.aspx	Report
ProRepPrepaidBalance.aspx	Report
ProRepScheduleAccountability.aspx	Report

ProRepWorkInProgress.aspx	Report
ProTimeCostTransactions.aspx	Report
PsnInvoiceForecastReport.aspx	Report
PsnReportBase.aspx	Report
SlsSalesAnalysis.aspx	Report
WizGraydonExport.aspx	Credit Check
XBRLInstanceDocuments.aspx	Country Specific

Appendix C. Case Study Results

Below all individual evaluation sessions are described. In the result tables the components are referred to by a number. The components belonging to these numbers are explained in the table below, the color codings show what components belong to the same sub category.

Component Number	Component Description
1	Outstanding items Receivable
2	Current Balance Bankaccounts
3	Contact Persons
4	Exchange Rates
5	Male/Female ratio
6	Birthdays
7	Number of invoice proposals
8	Written hours per project
9	Written hours per week
10	Billability
11	Revenue, costs, result
12	Total inventory value
13	Top items by revenue
14	Contractttypes
15	FTE Ratio
16	Employees per division
17	Number of proejcts per manager
18	Numer of projects this year and last year
19	Revenue per project this year and last year
20	Total amount in euros purchase invoices current year
21	Total amount of purchase invoices
22	Purchase invoices per GL Account
23	Purchase invoices per supplier
24	Returns
25	Sales orders: quantity
26	Sales orders: amount in euros
27	Number of active subscriptions
28	# of unprocessed entries

Wholesale user 1

Max number of components: 11

Percentage of correct suggestions (component): 81.82% (9/11)

Percentage of correct suggestions (subcat): 71.43% (5/7)

Component	Suggested by the model	Chosen by the subject
1	X	
2	X	X
3	X	X
4		
5		
6		X
7		
8		
9		
10		
11		X
12	X	
13	X	X
14		
15		
16		
17		
18		
19		
20	X	X
21	X	X
22	X	X
23	X	X
24		
25	X	X
26	X	X
27		
28		

	#	Reason
Extra Components:	6	Nice to know about birthdays
	11	Most important company figures
Did not pick:	1	Has no outstanding items, delivers after payment
	12	Inventory is not useful because there is no inventory. Buys when product is already sold.

Missing components: Urls to certain pages in the system.

Interesting quotes/findings: “Created own dashboard, my cockpit, your dashboard is similar but misses url’s to pages”

Wholesale user 2

Max number of components: 15

Percentage of correct suggestions (component): 80.00% (12/15)

Percentage of correct suggestions (subcat): 90.91% (10/11)

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3	X	X
4		
5		
6		
7	X	X
8		
9		
10		X
11	X	X
12	X	X
13	X	X
14		
15		
16		
17		
18		X
19		X
20	X	X
21	X	X
22	X	X
23	X	
24	X	X
25	X	
26	X	
27		
28	X	X

	#	Reason
Extra Components	10,18,19	Interesting to know about projects (note: User does not have project module)
Did not pick	23	only 1 supplier so per supplier is irrelevant,
	25,26	No outstanding sales orders because orders are handled immediately

Missing components: -

Interesting quotes/findings: Dashboard is more relevant to her than the standard dashboards

Wholesale user 3

Max number of components: 9

Percentage of correct suggestions (component): 66.67% (6/9)

Percentage of correct suggestions (subcat): 75.00% (6/8)

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3	X	
4		
5		
6		X
7	X	X
8		
9		
10		
11	X	X
12		
13	X	
14		
15		
16		
17		
18		
19		
20		
21		
22		X
23		
24		X
25	X	X
26	X	
27		
28	X	X

	#	Reason
Extra Components	22,24,6	Interested in this information
Did not pick	3	Contact persons are not stored in exact
	26	Only quantity of sales orders is important, not amount in euros
	13	Does nothing with items within exact, only changing their prices

Missing components: List of item prices

Interesting quotes/findings: -

Wholesale user 4

Max number of components: 8

Percentage of correct suggestions (component): 87.50% (7/8)

Percentage of correct suggestions (subcat): 85.71% (6/7)

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3	X	X
4		
5		
6		
7	X	X
8		
9		
10		
11	X	X
12		
13		
14		
15		
16		
17		
18		
19		X
20		
21		
22		
23		
24		
25	X	X
26	X	X
27		
28	X	

	#	Reason
Extra Components	19	Interesting to know about project revenues
Did not pick	28	Entries are not processed on request of the accountant

Missing components: -

Interesting quotes/findings:

-Use the standard dashboards a lot, but personalized like prototype would be appreciated

-Is a Professional services company but uses wholesale package

Wholesale user 5

Max number of components: 9

Percentage of correct suggestions (component): 88.89% (8/9)

Percentage of correct suggestions (subcat): 87.50% (7/8)

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3		X
4		
5		
6		
7	X	
8		
9		
10		
11	X	X
12	X	X
13	X	X
14		
15		
16		
17		
18		
19		
20		
21		
22		
23		
24		
25	X	X
26	X	X
27		
28	X	X

	#	Reason
Extra Components	3	Contact persons is interesting to see in dashboard
Did not pick	7	Does not want # of proposals but amount in euros

Missing components: invoice proposals amount in euros instead of # of proposals

Interesting quotes/findings: -

Manufacturing user 1

Max number of components: 10

Percentage of correct suggestions (on component): 80.00% (8/10)

Percentage of correct suggestions (on subcat): 83.33% (5/6)

Component	Suggested by the model	Chosen by the subject
1		
2		
3	X	X
4		
5		
6		X
7		
8		
9		
10		
11		X
12	X	X
13	X	
14		
15		
16		
17		
18		
19		
20	X	X
21	X	X
22	X	X
23	X	X
24	X	X
25	X	X
26	X	
27		
28		

	#	Reason
Extra Components:	6	Forgot colleague X's birthday today
	11	Important information to know about
Did not pick:	13	Only sell 2 products, so top 10 products is irrelevant
	26	Only number of outstanding sales orders is interesting, amount is predictable from this

Missing components: -

Interesting quotes/findings: -

Manufacturing user 2

Max number of components: 4

Percentage of correct suggestions (component): 25.00% (1/4)

Percentage of correct suggestions (subcat): 100.00% (1/1)

Component	Suggested by the model	Chosen by the subject
1		
2		
3		
4		
5		
6		
7		
8		
9		X
10		
11		X
12		
13		
14		
15		
16		
17		
18		
19		
20	X	
21	X	
22	X	
23	X	X
24		
25		X
26		
27		
28		

	#	Reason
Extra Components	25,11,9	Is not work/function related, just interested in this information about the company
Did not pick	20,21,22	Only wants to see purchase orders per supplier, is all the info he needs

Missing components: Returns per supplier, he is in quality assurance so interesting who the “good” and “bad suppliers are”
Interesting quotes/findings: -

Manufacturing user 3

Max number of components: 12

Percentage of correct suggestions (component): 10/12 = 83.33%

Percentage of correct suggestions (subcat): 8/10 = 80.00%

Component	Suggested by the model	Chosen by the subject
1	X	X
2		X
3		
4		
5		
6		
7	X	X
8		
9		
10		
11	X	X
12	X	X
13	X	X
14		
15		
16		
17		
18		
19		
20	X	X
21	X	
22	X	X
23	X	
24		X
25	X	X
26	X	X
27		
28	X	X

	#	Reason
Extra Components	2	Good to know current balance
	24	# of returns is interesting
Did not pick	21,23	Amount in euros and per gl account is sufficient

Missing components: Budgets

Interesting quotes: “The dashboard isn’t perfect but it’s a good start”

Manufacturing user 4

Max number of components: 15

Percentage of correct suggestions (component): 13/15 = 86.67%

Percentage of correct suggestions (subcat): 8/10 = 80.00%

Component	Suggested by the model	Chosen by the subject
1	X	X
2		
3	X	X
4		
5	X	
6	X	X
7		X
8		
9		
10		
11	X	X
12		X
13	X	X
14		
15		
16		
17	X	X
18	X	X
19	X	X
20	X	X
21	X	X
22	X	
23	X	X
24		
25	X	X
26	X	X
27		
28		

	#	Reason
Extra Components	7	Not relevant to this employee
	12	Not relevant to this employee
Did not pick	22	G/L account is not necessary if he can see purchase orders per supplier
	5	This is known because only 6 people work here

Missing components: -

Interesting quotes: -

Manufacturing user 5

Max number of components: 15

Percentage of correct suggestions (component): 12/15 = 80.00%

Percentage of correct suggestions (subcat): 8/9 = 88.89%

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3		
4		
5		
6		
7		X
8		
9		
10		X
11	X	X
12	X	X
13	X	X
14		
15		
16		
17	X	
18	X	
19	X	
20	X	X
21	X	X
22	X	X
23	X	X
24		X
25	X	X
26	X	X
27		
28	X	X

	#	Reason
Extra Components	7	
	10	
	24	
Did not pick	17,18,19	Does not work with projects

Missing components: -

Interesting quotes/findings : Users says they do not work with projects, but project related page views represent a great portion of this users activity log.

PSA User 1

Max number of components: 18

Percentage of correct suggestions (component): 14/18 = 77.78%

Percentage of correct suggestions (subcat): 10/12 = 83.33%

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3	X	X
4		
5	X	
6	X	
7	X	X
8	X	X
9	X	X
10	X	X
11	X	X
12		
13	X	
14		
15		
16		
17	X	
18	X	X
19	X	X
20		
21		
22		
23		
24		
25	X	X
26	X	X
27	X	X
28	X	X

	#	Reason
Extra Components	-	
Did not pick	5	Company is too small to worry about this
	6	Company is too small to not know this by head
	13	Only sells “Hours” so items are not that relevant
	17	Only 1 manager

Missing components: -

Interesting quotes: The suggested dashboard is similar to his ‘my cockpit’ dashboard

PSA User 2

Max number of components: 13

Percentage of correct suggestions (component): 10/13 = 76.92%

Percentage of correct suggestions (subcat): 8/8 = 100

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3	X	X
4		
5		
6		
7		
8	X	X
9	X	
10	X	
11	X	X
12		
13		
14		
15		
16		
17	X	
18	X	X
19	X	X
20		
21		
22		
23		
24		
25	X	X
26	X	X
27		
28	X	X

	#	Reason
Extra Components	-	
Did not pick	9	Hours are not monitored that strictly, only per project
	10	So billability is also not relevant
	17	There is only 1 manager

Missing components: -

Interesting quotes: -

PSA User 3

Max number of components: 13

Percentage of correct suggestions (component): 11/13 = 84.62

Percentage of correct suggestions (subcat): 7/8 = 87.50

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3		
4		
5		
6		
7		
8		X
9		
10		
11	X	X
12		
13		X
14		
15		
16		
17	X	
18	X	X
19	X	X
20	X	X
21	X	X
22	X	X
23	X	X
24		
25	X	
26	X	X
27		
28	X	X

	#	Reason
Extra Components	8	Hours are interesting, although company does not track hours in exact yet
	13	Top items would be interesting, but split on hours and products
Did not pick	17	Only 1 manager
	25	Only amount in euros is interesting

Missing components: -

Interesting quotes: Revenue split on hour sale, and item sale. Company sells both

PSA User 4

Max number of components: 14

Percentage of correct suggestions (component): 85.71% (12/14)

Percentage of correct suggestions (subcat): 88.89% (8/9)

9/10

Component	Suggested by the model	Chosen by the subject
1	X	X
2	X	X
3		
4		
5		
6		
7	X	X
8	X	X
9	X	X
10	X	X
11	X	X
12		
13	X	X
14		
15		
16		
17	X	
18	X	X
19	X	X
20		X
21		X
22		
23		
24		
25	X	X
26	X	X
27		
28	X	

	#	Reason
Extra Components	20	
	21	
Did not pick	28	
	17	

Extra Picks: 20, 21

Didn't Pick: 28,17

Missing components: Budgets

Interesting quotes: My cockpit, has seen it, but not clear how it works

PSA User 5

Max number of components: 6

Percentage of correct suggestions (component): 83.33% (5/6)

Percentage of correct suggestions (subcat): 100.00% (2/2)

Component	Suggested by the model	Chosen by the subject
1		
2		
3		
4		
5		
6		
7		
8	X	X
9	X	X
10	X	X
11		X
12		
13		
14		
15		
16		
17	X	X
18	X	
19	X	X
20		
21		
22		
23		
24		
25		
26		
27		
28		

	#	Reason
Extra Components	11	Interesting information on company results
Did not pick	18	Have a select number of customers, that all have multiple projects a year, but those can be short or long, number of projects is not as relevant as revenue

Missing components: Billability per Person.

Interesting quotes: -