

Increasing the efficiency of sales processes and reducing the interaction overhead with ERP-Systems by using mobile applications

BACHELORARBEIT

zur Erlangung des akademischen Grades

Bachelor of Science

im Rahmen des Studiums

Wirtschaftsinformatik

eingereicht von

Markus Drechsel, Lukas Inschlag

Matrikelnummer 1327557, 1325085

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Assoc. Prof. Dr. Dipl.-Ing. Hilda Telliöğlu

Wien, 10. Oktober 2017

Markus Drechsel, Lukas
Inschlag

Hilda Telliöğlu

Increasing the efficiency of sales processes and reducing the interaction overhead with ERP-Systems by using mobile applications

BACHELOR'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Bachelor of Science

in

Business Informatics

by

Markus Drechsel, Lukas Inschlag

Registration Number 1327557, 1325085

to the Faculty of Informatics

at the TU Wien

Advisor: Assoc. Prof. Dr. Dipl.-Ing. Hilda Tellioğlu

Vienna, 10th October, 2017

Markus Drechsel, Lukas
Inschlag

Hilda Tellioğlu

Erklärung zur Verfassung der Arbeit

Markus Drechsel, Lukas Inschlag
Vienna

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 10. Oktober 2017

Markus Drechsel, Lukas
Inschlag

Acknowledgements

First and foremost we want to thank our parents for the continued support during this time.

Secondly, we want to thank the company, where we have been able to conduct our evaluation. Especially, for providing detailed feedback and suggestions for improving the application. They really have been very supportive and had time for us when we needed it.

Last but not least, we want to thank Hilda Tellioglu for supervising this thesis. Particularly, for allowing us to independently write this thesis, but also for providing detailed feedback, when needed.

Kurzfassung

In den letzten Jahren wurden mobile Geräte zu einem festen Bestandteil des täglichen Lebens. Sie bieten viele Vorteile in jedem Aspekt des Lebens und funktionieren wie kleine persönliche Assistenten. Für die meisten Menschen ist das Leben ohne sie unvorstellbar geworden. Dennoch zögern viele Unternehmen immer noch, mobile Endgeräte in ihre Geschäftsprozesse zu integrieren, da dieser Trend noch relativ jung ist und bislang wenig Forschung in diesem Bereich durchgeführt wurde.

Ziel dieser Bachelorarbeit ist es daher, ein mobiles Endgerät in den seit Langem etablierten, standardisierten Vertriebsprozess, mit besonderem Fokus auf die Auftragserstellung durch Außendienstmitarbeitern, zu integrieren. Zur Bildung der theoretischen Grundlagen werden anfangs der standardisierte Vertriebsprozess und ERP-Systeme detailliert analysiert. Mit diesem Wissen wird eine Android App, mit Hilfe des User-Centered Design, entwickelt, welche die Android und Material Design Richtlinien erfüllt. Diese Anwendung wird dann mit den Methoden Cognitive Walkthrough, für die Designaspekte, und Keystroke-Level-Modell, um die Effizienz mit anderen ERP-Systemen vergleichen zu können, evaluiert. Die theoretischen Erkenntnisse werden dann analysiert und aufbereitet.

Das Ergebnis dieser Arbeit ist, dass die Effizienz des Auftragserstellungsprozesses, durch die Einführung einer mobilen Anwendung, um ein Drittel verbessert werden kann. Darüber hinaus kann die interne Kommunikation reduziert werden, wodurch die insgesamt Effizienz nochmals gesteigert werden kann. Schließlich kann gezeigt werden, dass selbst unerfahrene Anwender in der Lage sind, ohne großen Lernaufwand, die mobile Version zu verwenden.

Abstract

In the last couple of years, mobile devices became a part of everyone's daily life. They offer a lot of benefits in every aspect of life, functioning as a small personal manager. For the majority of people, life without them became unimaginable. However, many businesses still hesitate to adopt mobile devices into their business processes, since this trend is still young and not a lot of research has been conducted in this field.

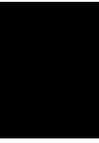
Therefore, the aim of this bachelor thesis is to incorporate a mobile device in the long-established, standardised sales process, with special focus on the order creation carried out by field workers. First, the thesis lays the theoretical foundation by analysing the standardised sales process and ERP-Systems. With this knowledge, an Android application is developed using User-Centered Design, following the general Android and Material Design guidelines. This application is evaluated using Cognitive Walkthrough for the design aspects and the Keystroke-Level Model to compare its efficiency to other ERP-Systems. The theoretical findings are then analysed and presented.

The results of this thesis are, that the efficiency of the order creation process can be improved by one third, through the introduction of the mobile application. Furthermore, internal communication can be reduced, improving the overall time efficiency even more. Finally, it can be shown that even novice users are able to easily adapt to a mobile version and use it with satisfaction.

Contents

Kurzfassung	ix
Abstract	xi
Contents	xiii
1 Introduction	1
1.1 Motivation and goal of the thesis	1
1.2 Structure of this thesis	2
2 Background and Fundamentals	5
2.1 General business processes	5
2.2 ERP systems	13
2.3 Synchronisation and data access back-end	22
3 State of the Art	29
3.1 Interface and Interaction design	29
3.2 The Android platform	32
4 Methodology	37
4.1 Literature Research	37
4.2 User-Centered Design	38
4.3 Evaluation Methods	41
5 Process Definition and Prototyping	45
5.1 Custom order creation process	45
	xiii

5.2	The synchronisation process	48
5.3	Creation of the prototype	54
6	Evaluation	77
6.1	KLM	77
6.2	Cognitive Walkthrough	91
7	Conclusion and Future Work	101
	List of Figures	103
	List of Tables	105
	Bibliography	107



Introduction

This section describes the process of how this work came into being and informs on the reasoning for why this topic has been chosen. Furthermore, the goals that this thesis wants to achieve will be presented.

1.1 Motivation and goal of the thesis

In recent years mobile devices haven been becoming more important than ever. Especially in the personal usage sector of electronic devices, the ability to take your device everywhere is becoming a real necessity.

More slowly, but with growing speed, the business section also starts to follow this trend. An important part, where this plays a huge role for companies, are the ERP-Systems. Where today, these are de-facto standards for managing the businesses data. Therefore, the need to be able to access this contained information everywhere is a striving factor for change in many companies.

While many large ERP-System vendors, like SAP, already invest heavily in building mobile version of its systems, these are only usable when a company already owns the, often rather expensive, licences for the desktop versions. Especially for small and medium-sized enterprises this is a real disadvantage, since they are often not able to afford these. Therefore, this thesis analyses the way such a mobile

ERP-System can be implemented and wants to show that it is also possible to use independent systems that are by far not as expensive. Furthermore, the prototype has been designed to be really flexible, allowing a vast number of databases to work with this implementation. This is achieved by relying on directly providing a database connection back-end, which includes an easy to configure translation service. Therefore the table structure, of the to be used database, does not have to be adapted to work with this application.

A further striving factor for conducting this research, is to show how mobile applications can actually improve a businesses process, like the sales process. Because with traditional systems, there is always the limitation of accessing and maintaining information in the company. Which has to be done on a desktop machine and this means being restricted to stationary work.

Particularly, in the sales field-workers job this is a huge disadvantage, since the field sales force still has to note orders on paper and later carry over these into the ERP-System. This is where this thesis' application comes into play, providing a direct access and thus removing this redundant intermediate step.

To sum this up, the goals for this thesis are, first to provide an analysis of current sales processes and secondly providing an adapted and improved version, enabled by the use of a mobile application. Furthermore, this application should be easier to use, thus being practical to work with in non-stationary work at the customer. Finally, this implementation will work offline too, therefore allowing to manage orders and check information even when no network connection is available.

1.2 Structure of this thesis

This thesis starts by introducing the topic of the usage of mobile devices in recent years. In chapter 1 it therefore describes the potential mobile devices provide in the business sector. Furthermore, the goals that this work wants to achieve are outlined.

Chapter 2 lays the theoretical foundation of this thesis. It therefore provides detailed information about business processes, focusing on the sales process. Following

this, the concept and usage of ERP-Systems are presented. Since for this thesis a functional prototype is implemented, the theoretical foundation for synchronisation with a database back-end is analysed too.

Chapter 3 looks at the state of the art information available for interface and interaction design. Since the developed application runs on Android, this operating systems' design language is described too, presenting the idea of Material Design and some of its included elements.

Chapter 4 describes the used methodological approaches. Starting at the literature research and going on with the, for designing this prototype, used method, called User-Centered Design.

In chapter 5 an for the mobile version adapted business process is presented. Following this, the synchronisation process is outlined. After that the actual designing and implementation phase, based on the User-Centered Design process, is described. This also includes the with the user groups conducted evaluation.

Chapter 6 finally evaluates the application with two methods. These are the Keystroke-Level Model, using the adapted Fingerstroke-Level Model, and the Cognitive Walkthrough.

Chapter 7 concludes this thesis, by giving a summary about the findings made. Furthermore, a possible future development for this prototype is outlined. Finally, the achieved results are compared to the, at the beginning described, goals.

Background and Fundamentals

To be able to design and develop business software, it is important to first understand the core concepts of business processes. A proper definition must be given to be able to comprehend all the requirements. This includes influencing factors, like environmental inputs, but also the targeted value. This works focus is on the sales process, for which a detailed description is provided. With this the core concepts and characteristics of an ERP system are discussed.

2.1 General business processes

Currently the organisational structure of businesses is shifting from a functional oriented to a process oriented structure. Since there is the general tendency that everything has to be faster, more efficient and customer-centric, every business has to follow this trend to stay competitive [Hof15] [HIP11]. To get a general understanding about what a process oriented structure is, it is important to define the fundamentals, as in what a general process is and its definition in the context of a business.

A process is defined as a series of actions or steps taken in order to achieve a particular end. This means that there is a designated start and end and a set of actions that are performed within the process.

A business process is built like this general process but evolved over the years into a complex, very well defined and standardised procedure. Hammer and Champy concisely define a business process as a set of actions that transforms one or more inputs into an output with value for the customer [Gad13].

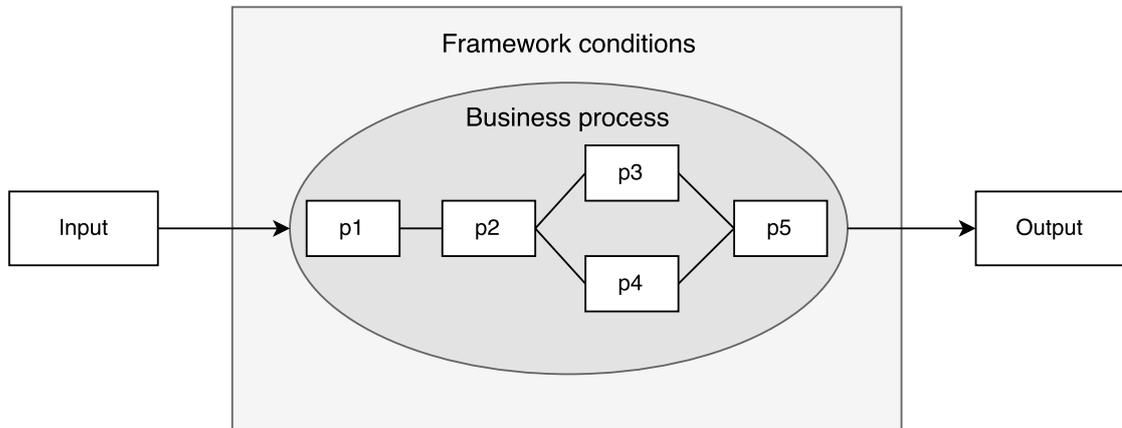


Figure 2.1: Simplified business process, based on [ACS96]

Figure 2.1 visualises this definition of Hammer and Champy. Every business process has an input. This input can be in material or immaterial form, for example raw material or information. It is then transformed over the course of the process into an output that has a value for the customer. The actual business process consists of one or more sub-processes that connect to each other, like p1 to p5 as shown in the Figure 2.1. These sub-processes can also consist of further sub-processes or finally elementary steps and be executed sequentially or parallel. The whole process is limited through the framework conditions. These are determined by the business policy. The framework conditions define a set of instructions on how to perform each step of the process and the order of execution [ACS96].

In a business, there are many different processes that follow this general structure, ranging from internal processes in different corporate departments to corporation-wide processes. There is a further differentiation of business processes into three subtypes.

The first subtype are the core processes. These are the processes that generate the most value for the business. Processes such as order processing, product development, production, distribution or service fall in this category. The second

subtype are the management processes. They are responsible for integrative interaction between all business processes such as strategy development and business planning. The last subtype are the support processes.

2.1.1 General sales process

Since this work focuses on the sales process a detailed definition of a standardised sales process according to Hofbauer and Hellwig will be given [Hof15]. It consists of 11 sub processes that will be described in the next section. The sales strategy, derived from the business strategy is the input of this process. It includes, for example the strategic planning, sales goals, available resources and competitive strategy, on which every further decision is based on. Therefore it also provides the framework conditions as stated previously in the general definition of a business process. The output of this sales process is value generation. This includes enterprise value, customer equity, as well as employee satisfaction.

The first sub process is organisation. Its purpose is to lay the foundation for a seamless sales process. Part of this process is the definition of the overall sales task, its analysis and breakdown into subtasks. Through task-synthesis these are assigned to corporate divisions, which form the organisational structure. At this stage, the sales channels are formed, consisting of field and office work as well as direct and indirect corporate sales. This is followed by the definition of the operational structure, which determines the workflow for each task and their order of execution. This sub process also includes the selection and vetting of employees as well as specifying possible training courses for them. The selection of the appropriate business information system is also part of this sub process. Nowadays a successful sales process without support of IT-systems is not possible. They play an essential role in every aspect of the sales process, like merchandise management, order processing, invoicing, information provision or customer orientation (CRM).

The second sub process is market planning. Since the foundation for the sales process is laid, the next step is to analyse the market and develop a market profile. This includes general market characteristics, like market size and trends, customer and competitive behaviour and the market positioning of the own business. Market

segmentation and target group definition enables a target oriented sales process. Furthermore, the allocation of resources, such as the number of employees and the sales budget, is planned based on the market analysis.

The third sub process is customer planning. The result of the market planning is now further specified by defining customer segments. Based on this, customer profiles can be created, which forms the base for all customer acquiring.

The fourth sub process is initial business contact. The first step in this sub process is to define which information the business wants to publish to potential customers. This includes the general range of offers and services, but also possible customer specific product upgrades and updates to awake interest in potential key accounts. The actual initial contacting can happen in two ways. If the business takes the initiative it is called active contacting. It has a lower acquisition success rate than the passive contacting, where the customer approaches the business. At this stage, the type for further communication is set. It could either be face-to-face, that a sales employee visits the customer, or semi-personal through voice-to-voice communication via telephone. The output of this sub process is a customer's request.

The fifth sub process is checking the customer's request. The customer's request is now precisely analysed. Its primary focus is to check if this request is profitable and whether its realisable or not. This takes the technical, economical and logistic aspect in consideration. This step is important since tender preparation is a costly process and does not pay off for every request.

The sixth sub process is tender preparation. The objective of this process is to offer the customer a tender that will lead to a conclusion of contract based on his preceding request. First, someone must be commissioned with the tender preparation. Usually, this will be done by a sales employee, but in larger scale it is possible that a whole team is responsible. Then, based on the customer's requirements, a solution is developed. This solution includes a functional specification and a draft of technical details. This is followed by a capacity alignment, schedule planning as well as a price determination for the tender. After the determination of delivery and payment terms, the tender must be confirmed by the management and is then

released to the customer.

The seventh sub process is preliminary clarification. If the customer is interested in the tender the next step is to clarify open issues through a detailed discussion. This should lead to a win-win situation for both sides and finally result in an agreement in form of a letter of intent.

The eighth sub process is negotiation. In this sub process the letter of intent is converted into a concrete contract. This contract includes for example the contractual object, norms and standards, dates, prices, terms of payment, packaging and transport, insurance and breaches of contract. This contract is negotiated and reworked until both parties are satisfied and every aspect is clearly defined. The output of this process is the conclusion of contract.

The ninth sub process is order management. Since the contract is successfully concluded, the order processing can begin. This also includes shipping preparation and delivery to the customer. Purpose of the order management is the assurance that the order is delivered as previously agreed upon in the contract.

The tenth sub process is after sales support. Since the delivery to the customer is finished, it is important to keep in touch with the customer. Further technical and mercantile support contributes a lot to long-term relationships and building a loyal customer base.

The eleventh and final sub process is sales controlling. The purpose of sales controlling is securing long-term success. The status quo is captured and thoroughly analysed. The results are then aligned with the initial objective. If there is discrepancy, measures for the next sales process are taken to further improve the degree of target attainment and thus secure long-term success of the business.

2.1.2 Order creation process

The last section gave a detailed presentation on what the entirety of a standardised sales process looks like. Since the focus of this work is about the creation of orders, by field workers, a detailed explanation on how the procedure of creating an order looks like and which information an order must contain.

First and foremost, there are different approaches to creating orders [JB04].

The best-case scenario for bigger sales companies is an EDI (electronic data interchange) supported order creation, since it requires the least amount of work and time investment of the sales department. It is based on a very close connection to its customers. The whole process of creating an order happens through EDI. The customer has online access to the merchandise information system of the sales company. Thus, all vital information is obtainable, like articles, their prices, discounts, current stock in different warehouses and much more. When the customer creates an order, the system automatically gets the customer information from the master data and no manual work is needed [JB04].

Although, this scenario is the minority. Usually, the order creation requires manual work and cooperation from the whole sales workforce, including in-house workers and field workers. This is a much more time-consuming and complex procedure. Furthermore, although customers have access to information about about the articles the sales business offers, this source of information is not always accurate, since it is mostly not up-to-date. The reason for this is that, these information are accessed through various sources like web services, newsletters or e-catalogs, which are not updated instantly.

The process starts with an order request from the customer. In-house workers check if the customer already exists in the master data. If not, master data for the customer is created. Following this, a meeting with the customer is arranged and the responsible field worker visits the customer at their desired location and timeframe. Together with the customer, the field worker checks the selected articles and informs the customer on new products, special offers and discounts on existing ones. If questions occur, that the field worker cannot answer, the in-house workers have to be contacted to provide information, since the field worker has no direct access to the merchandise information system of the company. Customer specific deals can be negotiated, for example, quantity discounts or discounts on special bundles. When the customer is satisfied, the sales employee can now summarise the order information. First, all the necessary customer information must be noted. This includes a unique identifier, that the order is assigned to the correct customer. Furthermore, a shipping date is noted, which defines when the order has to arrive

at the customer location. Payment information is noted and the in-house sales team checks the credit history of the customer. Afterwards, all articles will be written down in form of order positions. An order position specifies the article, the amount, a measure of unit (like a single unit up to pallets) and the negotiated discount [JB04][Gud99][Hac98].

The field worker now submits all this information to the in-house sales worker, which checks the information and creates an order entry in the system. If any ambiguities occur, the field worker is contacted for clarification or in the worst-case scenario even the customer. When the order entry is complete, the order creation process is finished.

This process heavily involves in-house workers, a lot of consulting, delays in time and possibilities for errors, which leaves great potential for improvement.

2. BACKGROUND AND FUNDAMENTALS

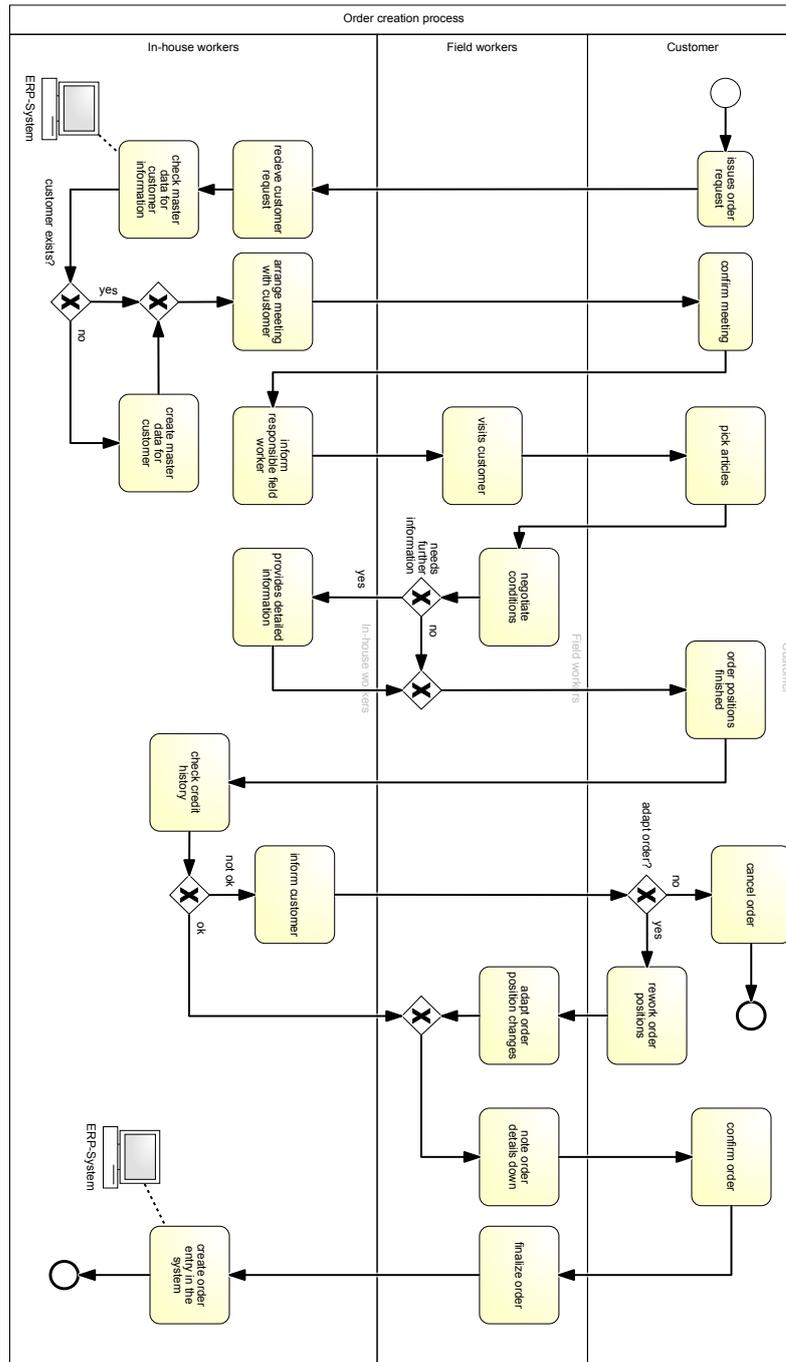


Figure 2.2: General order creation process, based on [JB04][Gud99][Hac98]

2.2 ERP systems

The usage of Enterprise Resource Planning (ERP) systems is common practice for nearly every company. Since the use of such business software enables them to manage their resources and to integrate the entire information flow. This includes, among others, financial, sales, supply chain and customer information. With all the company's resources bundled in one place the resulting software suit is relatively complicated to maintain, develop and especially to adapt. Thus, every ERP system includes its own set of implemented business processes, which are fully functional and ready to use. These allow to partly automate and to monitor critical steps of each business process [ABES14] [Dav98].

An example for this would be the liquidity management in an ERP system. Especially in larger enterprises the number of payments by far exceeds the financial accountants overseable amount. This means that an accountant would not be able to easily and in real-time know the value of the circulating assets. An ERP system on the other hand manages all this data and thus can keep track of the current assets and warn the account about possible liquidity problems. By analysing the data over time the system is even able to predict future liquidity. With this a company's liquidity can be better secured and critical situations prevented.

2.2.1 The value generating system

To better understand the value that an ERP system generates for a company the fundamental characteristics of a system must be considered. A system generally can be defined as a collection of interrelated parts that serve some purpose. Furthermore, any changes to one of its parts results from or leads to a change in some other part [Ack71]. This directly correlates to the usage of the ERP system as a holistic view on all the enterprise's information. Because a company consists of many connected entities, which all together form a value generating business. Thus, it is important that the fundamentals of how all parts work together are well understood and properly defined.

Basically, a system can be modelled as seen in Figure 2.3. It takes inputs and produces outputs through its processes. These are controlled by varying mechanisms,

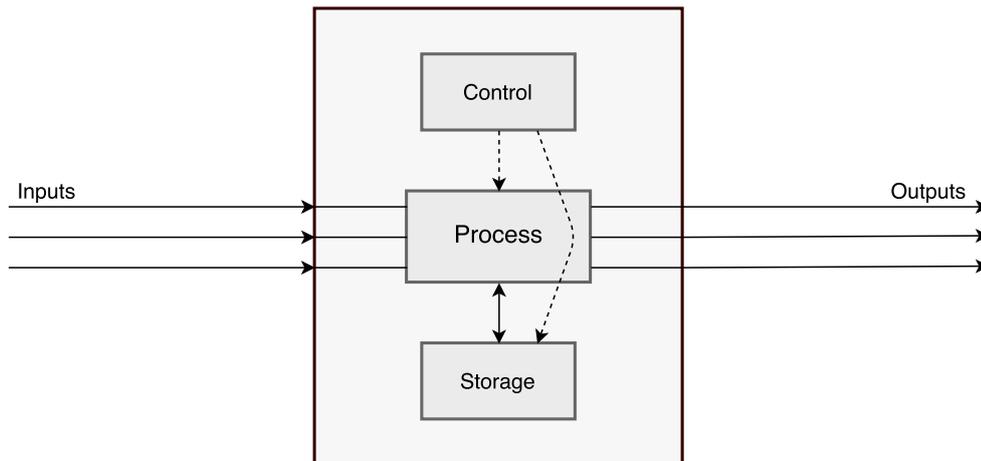


Figure 2.3: A general model of a system, based on [CC01]

which assure the proper functioning. Any needed information and parameters are stored in a storage, which can be accessed by the processes [CC01]. In the business context, such a system has the objective to generate value for the company. Value is a broad term, which primarily means the output the system generates but also the efficiency of its processes. The efficiency is measured by looking at the input to output ratio. This means that a company is more efficient if it generates more output with the same input [Wat12].

In a company the inputs are separated in categories like materials, energy, labor power, information, decisions and money [CC01]. Thus, any output and therefore the value for the business consists of an optimal combination of these.

2.2.2 The characteristics of ERP systems

ERP systems are typically separated into modules, representing the business entities. Grammer describes this separation as the four classic pillars that add up to an ERP system. Such as that the complete functionality is made of the modules purchase and logistics, human resources, finances and sales [Gra11]. This structure is represented in Figure 2.4.

The modules are part of the application domain, which is located on top of a database management system (DBMS). This DBMS manages and provides the shared data for the ERP system. Through this, the various parts can communicate

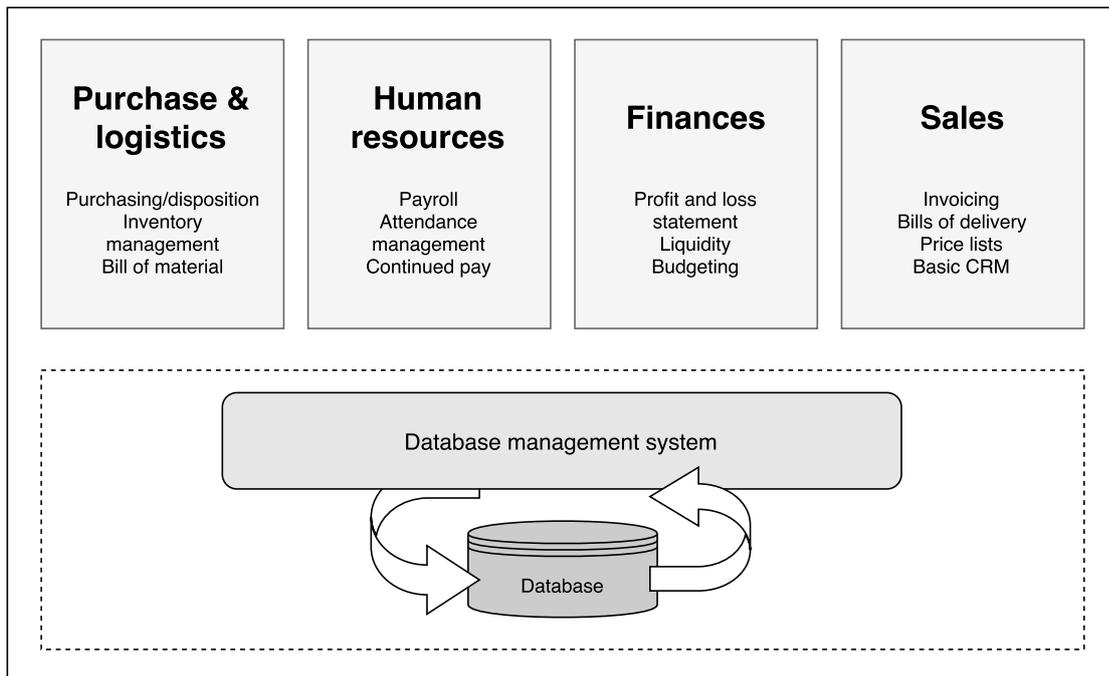


Figure 2.4: ERP base modules, based on [Gra11]

with each other and fully support the business processes.

To achieve this tight interlinking a certain level of integration is needed. Specifically, data, function and process integration is what allows this connection of all the various business aspects. This is also what distinguishes ERP systems from the usually used, isolated and function specific programs.

Data integration means that one shared database is used across the whole system to allow different functions to communicate with one another. Typically, this is achieved by using a relational database, like Oracle or Microsoft's SQL [O'L04]. These store the data recorded in the daily business transactions, but also merge it into an enterprise-wide operational data set. This is then made available by the DBMS in various formats to allow a seamless integration into the application domain [CTC05].

Full data integration reduces the work load since data, that is needed in various functions, does not have to be manually entered for each. It also decreases deviations or errors, which potentially could lead to inconsistent data, which could cause

serious problems. This aspect is especially important in the finance department, where every mistake could lead to a loss of money or even penalties. Furthermore, it decreases the time needed for data movement, resulting in nearly instant available information across the whole company [Mei11].

To achieve data integration multiple concepts are available. Basically, there are two possibilities varying in complexity and integration capability. The first option are data exchanges. These are simpler to implement and use export and import mechanisms, or interfaces. Data will be formatted (typically CSV or XML), exported and later imported by the requesting application. While this kind of data integration is simple to achieve, it is also no real data integration. Since the data is not really shared and during the exchange errors can occur, which would lead to inconsistencies. Moreover, this type heavily relies on formatting, which means that every application wanting to use the data needs to support the used format. The second option is the fully integrated data storage. Here, every application accesses the same data and thus the information is available in real time [Mei11]. While this type is closest to full data integration it still has drawbacks. Because to be able to access the same data, a shared database is needed. This leads to a tight coupling of application and data storage [Gro10].

For ERP systems that provide all modules this is usually no problem, since all components are developed and integrated by one vendor. However, most ERP systems do not fully meet all requirements of a company and thus third party applications are needed. Integrating those into the shared data system is a complicated process with high programming effort, or sometimes even not possible.

Functional integration means that each task is executed in the systems' domain. Furthermore, all tasks are harmonised to one another to make sure that the results of one step can be properly interpreted and used by the following one. This coordination of tasks has to take place over several levels and operating units. Resulting in both horizontal and vertical integration, which can be seen in Figure 2.5.

The three layers shown can be described as, starting at the top, the strategic, administrative and operational parts of a company, which together add up to the organisational objectives [SWW02]. The management, mostly consisting out of the

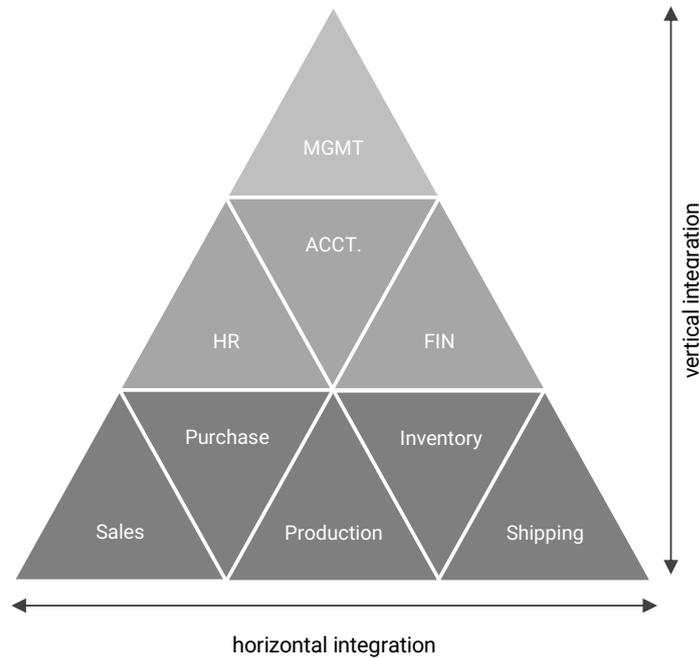


Figure 2.5: Integration in a hierarchic structure, based on [Mei11]

executive board, formalises the rules and the company strategy, and controls all functional entities. The administrative and the operative layer comply to these rules and act to achieve the strategy. ERP systems act out the horizontal integration by fully integrating the business processes, which mostly consist out of entities in the operative layer. If this is the case for all modules of the application domain, process integration is achieved. Process integration is thus an extension of functional integration by introducing workflows, which link different parts of the system together. With this it is possible that one module requests functions of another to get information or start following steps. If the ERP system is also linked with, or includes, human resource or management systems, vertical integration can also be achieved. This leads to a system that fully integrates all organisational objectives [Mei11].

2.2.3 CRM and SCM integration

Aside from ERP systems, customer relationship management (CRM) and supply chain management (SCM) systems also play a critical role in further extending the corporate value of using technology through enterprise systems. Investing in these has been found to affect both the financial and the performance side of a company positively [HSS07]. While each system on its own shows potential benefits for the company, only a thorough integration of both the IT system and the business strategy generates the full possible value [ASS02].

In the context of sales operation, the integration of CRM and SCM systems helps to further improve the service. Since information about customers is at all time accessible and can be automatically provided. This is especially useful for the field sales. By integrating these systems into mobile devices, the field workers can easily access their customer's information. Which helps them to provide better service not only before making a sale but also in the after-sales area. Furthermore, these systems can provide a benefit to the customers by making information, on for example their ordered items, directly viewable [HSS07].

Aside from providing information for the customer or the sales clerk, SCM systems also improve the operational and business planning capabilities. By managing the company's products and logistics with such a system the capacity planning can be partly automated. This allows for a faster reaction to supply and demand changes, making requested goods more readily available. Overall SCM systems increase the productivity by reducing the planning overhead, which allows for a more efficient inventory management. This reduces the operational costs by lowering the general inventory because goods can be ordered more on demand. Furthermore, the used capacity planning algorithms can reduce the overall order-to-fulfillment cycle time, resulting in better service productivity and customer satisfaction [HSS07] [ASS02].

CRM systems are not only used for accessing information but also to provide better controlling and decision-making abilities. The goal for using CRM systems is to build long-term relationships with customers and to increase customer satisfaction. This is achieved by learning to understand what is most important for each customer [HSS07]. Moreover, CRM allows for analysing the customers' habits and

using this knowledge to provide service beforehand. An example for this would be to analyse the customers' order history, recognising patterns and suggesting offers on most requested products. With this, the SCM can be further improved by preparing goods that will likely be requested soon, as identified in the customers ordering patterns [ASS02].

CRM and the gathered information is also beneficial for the companys' sales management since it allows to recognise market trends and shifting the enterprises mid- and long-term focus in the right direction. Furthermore, it provides the necessary tools for the manager to influence the behaviour and outcomes of marketing and sales personnel. Which is made possible by analysing the sales unit effectiveness and implementing improvements in the sales process [BCP01].

2.2.4 Mobile ERP systems

In the last section, Enterprise Resource Planning systems and the benefits of their usage have been thoroughly presented. Despite all their advantages they are still mostly restricted to the location of the enterprise. They are only accessible through inhouse access points and therefore offer no mobility and flexibility, which is needed in many departments of an enterprise, for example the sales department. As a result for these needs, there is an upcoming trend for mobile ERP systems. More and more is invested in research and development in this area, since being able to retrieve vital information for customers at any point, without a lot of detours and time requirements, would multiply effectiveness and customer satisfaction [DJ03].

The benefits of using mobile applications to interact with customers and providing a mobile access point for enterprise workers appeals to a growing amount of companies. For example, the Cerner Corporation, a major developer of healthcare information systems, has already developed a wireless solution for hospital staff. This allows to retrieve important clinic information at any time and not being bound to specific access points. This principle of instant access to information when needed has not only benefits in the field of health care, but can also be applied to any enterprise department that involves direct interaction with customers. Further case studies in large companies showed very positive results in such departments caused by the

adoption of mobile solutions [NSS05][BP08].

A study about the value of mobile applications for companies has been done by Nah (2005), that lists the benefits and improving factors, as well as risks and critical areas, which impede the implementation of such as system [NSS05].

The major benefits are mobility and flexibility. The traditional ERP solutions (except for a few like SAP) do not provide a mobile version to access the system. This hinders especially field workers in business areas like sales operations, but also influences the efficiency of in-house workers. Since they have to directly provide information to the sales workforce. Currently, the workflow of sales processes in many sales oriented businesses works as follows: The field worker notes the customer details and orders on paper and later on, this information gets updated into the ERP system by an office worker. This means there is redundant work, a substantial delay in time and more possible points for errors and failure. All those stated points can be removed by implementing a mobile ERP interface or access point. The field worker could immediately enter the acquired data into the system, which would eliminate the delay in time and redundant work. Studies have shown that this improvement over the traditional way of communication and coordination between field and office workers saves up to an hour a day per employee [NSS05][THK03][BP08].

A further point for mobile ERP systems is the increase of customer satisfaction. Up to date information, which is accessible for the field worker allows them to supply the most recent information directly to the customer. This increases quality of service immensely [NSS05][BP08].

Whereas a big antagonist to adopting a mobile ERP system is the security aspect. This is an important point of interest for every enterprise, but less of a struggle to implement and maintain with the usual inhouse ERP system. ERP systems usually are only accessible from inside the enterprise. Therefore, any intruder can only access information from inside the business. This makes it generally more difficult to access confidential information unauthorised. Furthermore, by implementing a mobile solution, the risk of loss or theft of a mobile device, that grants access to business information, is high. Another vulnerability is the data transmission, since

sending data over the internet, instead of just the intranet of the company, leaves room for intruders to intercept or even tamper with information. Aspects like a secure authorisation and encrypted data channels need to be taken in consideration [NSS05].

Another critical point is the general employee acceptance. It is common that employees prefer the status quo of a functioning system. Even if it has flaws or some inconveniences there is a workflow and tools they are accustomed to. Not everyone will embrace the introduction of new technology. Despite possible benefits, the learning process to handle new technologies and different workflows is intimidating and discourages people (especially elderly employees). Ultimately, a change process to include a mobile ERP system will fail due to this rejection. Therefore, elaborated plans to introduce the technology and provide courses for employees are key to success [NSS05].

A further aspect that influences the decision whether to implement a mobile ERP solution or not is its' cost. There are high initial payments for the new hardware, software and its adaptations to suit the needs of the enterprise, the extra service fees for wireless access and eventually training the employees. These need to be balanced with the predicted benefits a mobile ERP system will end up having for the enterprise. There is no general statement to whether businesses should all implement a mobile solution. This cost-benefit analysis needs to be done for each company individually [NSS05].

There are also very substantial differences in what an ERP system on a mobile device can do in comparison to the normal desktop environment.

First, the interaction with the system changes drastically. Normally an employee has access to a keyboard and a mouse, but with mobile devices there is only touch interaction available. Of course, smaller additions like a miniature version of a keyboard can be added, but this would make the device more stationary. This ultimately defies the reason to use a mobile device in the first place. As a result, the representation of information and content, as well as entering data needs to be adapted properly.

Furthermore, the screen size is a big issue. When using a tablet, the conventional

screen size is around 8 to 10 inches in screen diagonal. This is less than half of the normally used monitors in a desktop setup. In addition to that, all interaction takes place through that screen via touch. Usually every view of an ERP system is fully packed with information, which simply cannot be properly displayed on mobile devices due to their smaller screens. This means data and their presentation needs to be efficiently reduced to the most essential information and workflows need to be redesigned in more sequential steps to achieve the same result.

Another drawback of mobile devices is the lesser computing power, smaller storage and restriction through having a limited battery life. Therefore, it is important while developing a mobile solution to take all these restricting aspects into consideration. Maximising battery life is a huge issue, since it is not practical having to recharge the device several times throughout the work day. Connectivity is another very important aspect. Since there is no guarantee that the device is always and in every location connected to the ERP system in the enterprise, the application must work offline as well as online. These points are essential to make mobile devices competitive or even used in general [BP08].

2.3 Synchronisation and data access back-end

To be able to provide a mobile ERP system environmental factors have to be taken into consideration. These include the restriction of mobile devices in both hardware capabilities and network reliability [LKC04]. Therefore an efficient synchronisation mechanism has to be established. Thus, first the conventional way of accessing data in an ERP system is explained. After that, an adapted process is presented, which tackles the limitations of mobile devices.

2.3.1 Data exchange in in-house ERP systems

In-house ERP systems are used in a company intern network and run on desktop machines or notebooks. By being on a local network, internet speed is no constraint, which allows for a direct data exchange with the database. This means any changed data records are nearly immediately available.

While immediate data access is important, providing data consistency and with

this complying the ACID (atomicity, consistency, isolation and durability) is the priority. Therefore, to be able to guarantee data consistency usually one of two strategies is used.

The first strategy is to lock the accessed part, for example, an item master record, over the complete course of editing it. This approach is typical for distributed database management systems (DBMS). With this, it is guaranteed that there are no inconsistencies since only one user at a time can manipulate this record. Though this has the disadvantage that no other user can edit entries of this section at the same time. Especially with components of the ERP system, like the item master, which consists of much different information, this can be inefficient. In the case of the item master one user could want to change the item characteristics while another user changes the inventory location information. With this strategy, simultaneously working on this item would not be possible. Which leads to a higher effort of coordinating the users' workflow in the company [OV96][OV11].

Thus the second strategy takes a different approach. It sticks more closely to the original ACID principle, by implementing the relaxed ACID strategy. The goal here is to enable data consistency by guaranteeing a consistent state at the start of and when committing each transaction. Instead of locking the part to one user, using the isolation principle, all updates are hidden from the other users until the update is committed. Such an approach works by ordering the execution of transactions by timestamps. With this, all updates are done in sequence on the primary node, which in most cases is a centralised database server. Therefore, all users always work with a consistent state and there is no need for conflict resolution [Fra06][OV11].

However, since mobile devices differ greatly in both network availability and reliability, compared to desktop machines, a different solution has to be implemented. Instead of immediately providing changed data, the records are only fetched in a predefined interval. This is done because a continuous connection to the database cannot be guaranteed in all cases. While this approach solves the problem with not being able to constantly being connected to the database, it comes with its' own difficulties. Because data consistency is still the priority, conflict resolution strategies have to be used [LKC04].

Model representing the three strategies

To better understand the differences of these approaches the three strategies are modelled in Figure 2.6, representing a case where two users work on the same section at the same time. Here, the procedure of editing information is shown, focusing on the transition between a consistent and inconsistent state. Furthermore, different points in time are marked, indicated by $t[1-4]$. An example for this would be editing information on the item master, like described in the paragraph about the first synchronisation strategy.

For all three strategies starting the process, both users' views on the ERP system show a consistent state. In this example both users want to change multiple information. The first user changes the items' name, weight and price. The second user changes the items inventory location, safety symbol and the condition.

Strategy 1: Blocking

At time t_1 the first user starts to edit information of the item, thus his view represents a inconsistent state. This results in a lock of the whole items' section. Now he is able to do all his changes. At t_2 the first user is done, the lock gets removed and the changes are published, leading to a consistent state. At t_3 the second user can start to edit the item, once again locking the section. This time the second users' view is inconsistent and once he is done, at t_4 , his changes are published and his state is consistent again.

Strategy 2: Relaxed ACID

At t_1 the first user changes the items' name, once he is done this change is already committed. Here, no section lock is used and the second user does not know about the activity of the first. Thus, the second user is already able to do his first edit, changing the inventory location. Once he is done, his changes are also immediately published. Meanwhile, the first user also changes the weight and at t_2 both users' views are consistent again. This procedure is repeated between t_3 and t_4 . Here, the interesting change is, that both users really simultaneously change information, which is possible since all changes happen on the atomic level (in this case single information fields).

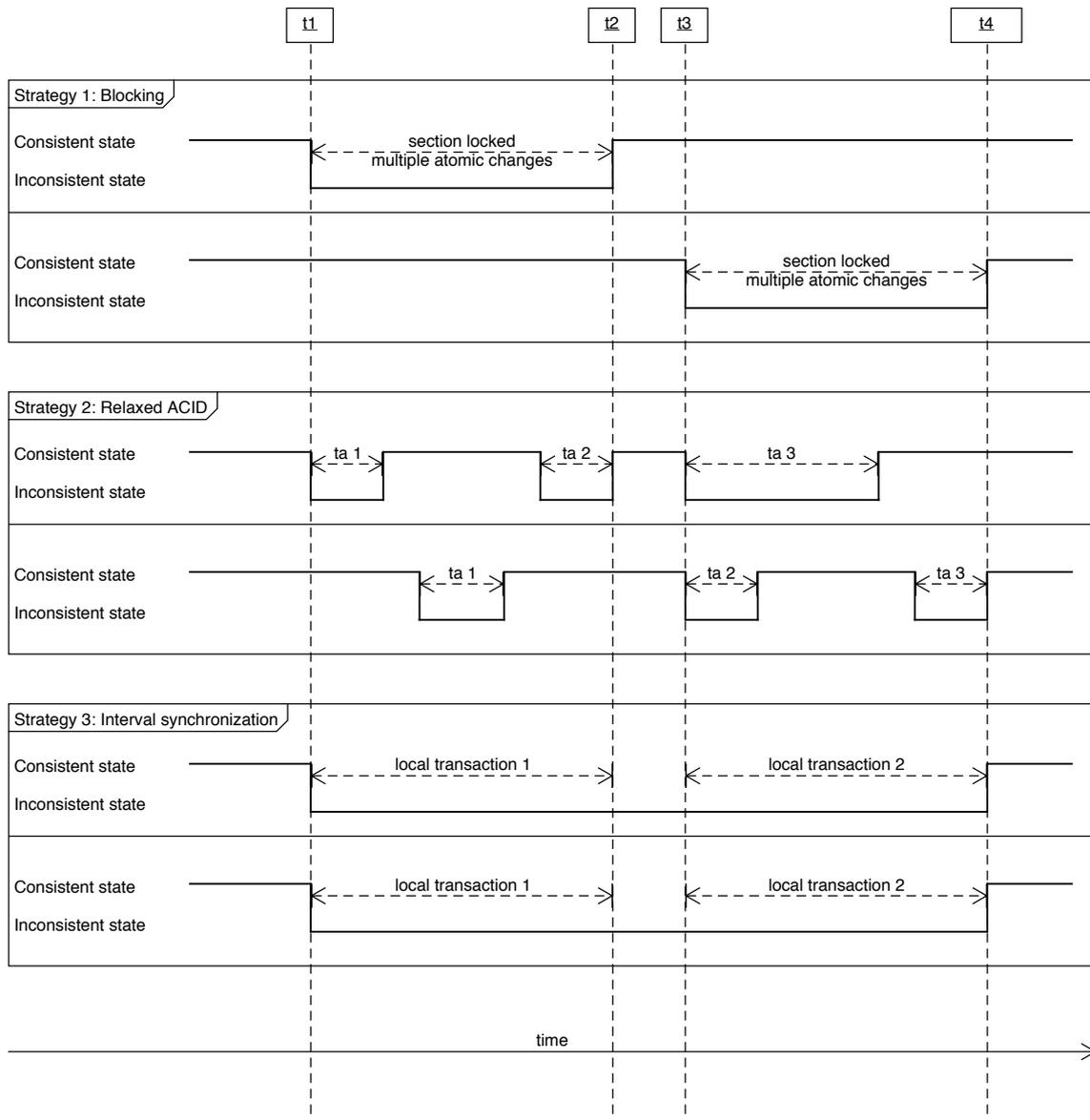


Figure 2.6: Model representing a case handled by the three strategies, based on [OV96][OV11][Fra06][LKC04]

In the case that both users would change the same information, the timestamp-property for each transaction (ta) would be used to decide the order of updating the information.

Strategy 3: Interval synchronisation

At t_1 both the first and the second user start to edit the item master. Between t_1 and t_4 all their changes are stored as local transactions. Which means that their changes are only stored in a local database, which represents a copy of the master database. This differs from the other two strategies, which directly access the master database. Therefore, since each user has its own copy, no locking mechanisms or timestamps to order and differentiate the users changes have to be used.

At t_4 all changes are done and to return into a consistent state the changes have to be merged into the master database. Because with this approach the local data is only synchronised in a predefined interval, this means that there could be conflicts with existing changes at the master site. Therefore, conflict resolution strategies are needed. Typically one of three approaches is used.

The first one is the "master data first" approach. This means that if there are conflicts, which are existing changes on the master site that occurred after t_1 , the existing changes are picked and the local adoptions are discarded.

The second one is the "local data first". This works similar to the "master data first" idea, just that the local changes are used and existing changes on the master site are discarded.

The third one is the "merge changes" approach. Here, all conflicting changes between the local and master site are taken and compared. A user or a set of rules then decides which changes are picked and which are discarded. While this strategy is the most complex, it mostly provides a better end result, since for each atomic change a comparison and decision is made.

2.3.2 The mobile synchronisation process

For designing the mobile synchronisation process, various aspects have to be taken into consideration. As discussed in Section 2.3 mobile devices cannot rely on a constant connection to the in-house ERP system. Therefore the synchronisation process has to be designed in such a way, that it supports only exchanging information in a certain interval. Because of this constraint the typical eager replication an in-house ERP system uses is no option. Instead they rely on a type of lazy

replication, which uses timestamps to decide whether to update the master data or to discard the commit (if there is already a newer commit) [GHOS96].

To be able to achieve this, it is necessary to locally store the changes until the next synchronisation takes place. This is either done by storing the updates directly as transactions or by using a local database to store the changes.

Since using a local database allows to more easily access and work with local changes, it was decided to use this, instead of storing the transactions. Therefore, the master-slave or two-tier model is used. This means that on one mobile node two separate databases exist. One only consists of exact replications of the last available master data and the other one stores all local changes [GHOS96][LKC04].

Lazy replication of master data

To be able to have an exact replication of the master data, the lazy replication scheme is used. Though, this only means that the local state matches the master state exactly after the synchronisation took place. Typically this is sufficient, since most use cases, where the ERP system is being accessed over a mobile node, do not depend on always up-to-date data. Especially when implementing a mobile solution for sales workers, where only article and customer information are accessed, this is the case. Because both article and customer data are of the core data type. Which are data records that typically do not change regularly. While for example construction data records, like work steps, update frequently.

Lazy replication therefore allows to update the master data in a defined interval, requesting all available changes that occurred after the last synchronisation [WPS⁺00].

Two-tier replication with a local update scheme

The two-tier replication approach supports the idea of the lazy replication of master records, because it is based on the separation between a master and a local version. Where the master version represents the most recent values, available from the master site, and the local version all the local changes. Furthermore it distinguishes between two types of transactions. First there is the base transaction, which only works on master data and produces new master data. This type is used to publish

2. BACKGROUND AND FUNDAMENTALS

local changes to the master site and to update the local master replicas. It therefore is a transaction that occurs between the mobile and the base (master) node. The second type is the tentative transaction, which only works on the local data and produces a new local version [GHOS96].

State of the Art

3.1 Interface and Interaction design

A very important part in creating a mobile application is the interface and interaction design. The design process and important design factors for mobile applications are different from desktop-pc applications. Many aspects have to be taken into consideration, like the different ways of interacting with the application, the screen size, the environment where the application is used and much more. The next section presents the most important design rules that need to be considered when creating a mobile application, resulting in a high level of usability.

Usability is a key factor for users and their loyalty towards the product. Usability is defined as how efficient, effective and satisfying a certain user can work with the mobile application. Providing usability is a win-win situation for the user as well as the developer. If the interaction with the system is more convenient and easy for the user, they will likely use it more often and recommend the application to others [HV15].

When designing a mobile application, there are general rules for the placement of user interface elements available. The first rule is alignment. User interface elements that share the same context should be horizontally and vertically aligned with each other on an invisible grid. This creates structure and clarity for the user. The next

rule is proximity. Elements that are content-related should be grouped to create a cohesive unit. Following this, the distance between not content-related elements should be apparent and clear. Furthermore, creating contrast is crucial. This can be achieved through differentiation in color, shape and size. Contrast is useful to illustrate the structure and hierarchy, for example to distinguish headers from their related content. The last general rule is repetition. The alignment, proximity and contrast of elements should repeat itself throughout the whole application. This creates consistency and therefore familiarity for the user, which improves usability [Hen15].

Colors are another very important aspect in the design process. When picking a color scheme for the application, a few aspects have to be considered. First, the meaning of color differs depending on the cultural background of the user. Colors evoke different feelings in users with different cultural backgrounds. This is especially important when indicating system states through colors. Furthermore, colors should be used sparingly and consistent. Using an overwhelming amount of colors only distracts and confuses the user. Moreover, the harmony of the colors in the scheme is crucial. When colors are used that do not match, the application will not appeal to most of the users and distract them. Currently, there are many different tools available on the internet to determine colors that harmonise. For example, the color tool of Googles' material design. Before these online tools became predominant, other instruments could be used to determine harmonising colors, such as the color chart introduced by Harald Küppers [KG06][Kue78][Hen15].

Text is, besides graphics, the main communication medium with the user. Therefore, it is important to make text as easy to read and understand as possible. The main way to impact the appearance of text is by changing the font. There is a vast amount of fonts available, but studies show that not every font is appropriate on mobile devices. One particular study that was conducted by QUALCOMM showed, that many factors like character shape and spacing, word shape, line spacing and proportion are crucial aspects for readability of a font on mobile devices. Therefore, big companies established their own optimised fonts, for example Google's "Roboto" or Apple's "San Francisco". Furthermore, text can be manipulated through size, color and boldness to create contrast and hierarchy between elements like headers

and their related contents [BOB05].

Icons are also a very substantial topic of mobile application design. There are various factors that need to be considered when deciding for icons. Generally, icons are small graphics that convey a certain functionality. There are several aspects an icon needs to comply to be more useful than a textual description. Familiarity is the first point. It describes how frequently the icon is seen, more precisely, how likely the user will recognise the functionality of the icon. Thus, the higher the familiarity, the shorter the required learning time. The next point is clarity, which deals with how legible the icon is. This includes the structure and shape and must be adapted for different screen resolutions. Furthermore, icons need to be kept simple. Unnecessary complexity and details only irritate the user. Icons also need to be consistent throughout the application.

Usability is strongly impaired, if different icons for the same functionality are used. Overall, icons should be understandable without added texts as explanation, but in some cases, when the functionality is more complex or abstract, a descriptive label can clarify the function. Furthermore, the choice of colors should always regard the background where the icon is located, to generate a high contrast and make the icon more apparent. The icon size should also be consistent throughout the application. Another important point is, that there are already established icons for certain functionality, used by other applications or the operating system itself. These should be used for similar functionality in the application, like the navigation menu icon in android (a stack of three horizontal bars) should be used for opening the navigation menu in the application. Misusage of these icons will likely create confusion for users, leading to bad usability. Regarding all the previously stated points, the developer can design custom icons, or choose from widely elaborated and refined icon libraries, such as the icons from Google's Material Design [BT93][Gal07].

Another very important aspect is feedback. The user should always immediately get feedback from the system after executing an action. Feedback can take various forms, like obvious content changes, a text notification that the action was successful or has failed, or a loading indicator, if the systems' response has longer processing times. Feedback should always be in the "users' language". That means, if for example an error message is displayed as a text notification, it should explain the

problem that occurred in a way, that novice users can understand it and if possible offer a solution for the problem. It should never just display the error code from the system. Without feedback, it is difficult for the user to recognise if an action was processed by the system. This has a negative effect on users' effectivity, efficiency and satisfaction [Hen15].

The environment where the application is used is also an important factor. Knowledge about the environment influences most of the previously stated design aspects. For example, if the intentional environment where the application is used is indoors, different colors, intensity and amount of contrast is needed than when it is used outside, due to different lighting and constantly changing weather. Following this, it is also a difference if the application is intended to be used while moving or while being stationary, which would influence the placement of interface elements and the general interaction, since the interaction with the application is mostly restricted to one hand when moving [dSC08].

The last crucial aspect is the restriction of available space on mobile devices, due to their smaller screen size. This directly impacts the amount of information that can be displayed at once. As a result, information must be reduced down to the most important ones and workflows from desktop systems need to be adapted to fit the screen size and interaction method. A general rule of thumb is, that design is not perfect, when nothing more can be added, but rather when nothing of the current can be left out [Mos12].

3.2 The Android platform

3.2.1 A brief history on Android Design

Android was founded to support a vast variety of different device form factors and input methods. To be able to achieve this, the early implementations focused mostly on the technical back-end. This led to an easy to use framework for developers, enabling them to build applications for numerous devices, without the need to adapt the code for each. The idea was to have a similar approach like fluid web design, which adapts to different form factors on its own. Though, since the

Android developing team focused much on the technical background, the interface design did not get much attention. This led to a blunt and mostly inconsistent visual experience. This only started to change in 2010, when Google started to focus more on the User Experience (UX). Starting with the Android version 3.0, Honeycomb, they introduced its first design language, called Holo. Over the course of the version 4.0, Ice Cream Sandwich, and the following, they were able to bring a more cohesive experience, focusing on simplicity and consistent elements. While this was a huge step forward, many still criticised the lack of a visually pleasing design.

Thus, with Android 5.0 Google released Material Design, its own design language. But Material Design is meant to be more than just a style of design, it intends to rethink the way users interact with Android, introducing a whole set of design principles.

Going forward, all applications should adapt to these ideas and even older versions were able to use some of these ideas, by relying on special support libraries, bringing some of these new features to older Android versions. [Cli15]

3.2.2 Material Design

Google describes its Material Design in this way: “A material metaphor is the unifying theory of a rationalised space and a system of motion. The material is grounded in tactile reality, inspired by the study of paper and ink, yet technologically advanced and open to imagination and magic [Incb].”

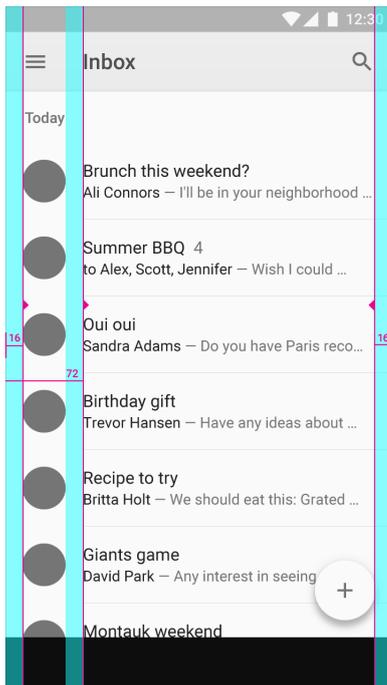
With this, they intend to relay the idea, that the interface of Android applications should be based on real paper and ink, but not limited to it. This idea is heavily supported by the introduction of elevation, resulting in shadows, as a main design element. This generates a visual hierarchy, supporting the z-axis, and better represents the way real elements are stacked over one another. Furthermore, with Material Design, standardised spacings, shown in Figure 3.1a, have been introduced, which are automatically applied to all by Android provided elements. This helps to create a more consistent experience across the whole system.

With Material Design, a few new elements have been added. One major interaction

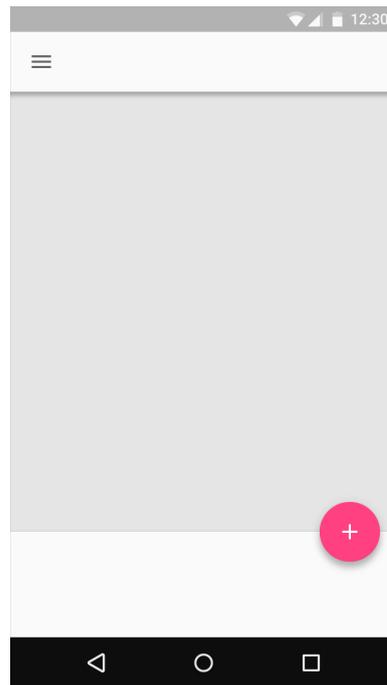
element added, is the Floating Action Button (FAB), shown in Figure 3.1b. In most applications this has become the main interaction point on a given screen. This is supported by its visual outstanding representation, with a bright color as background and more elevation compared to other elements. The FAB mostly is used for primary actions, for example the Gmail application uses it for opening the screen for composing an email.

Another important change, that came with the usage of this design language, is the better support for providing feedback to the user. Therefore, two new elements, the Toast and the Snackbar, have been added. The Toast, shown in Figure 3.1d, is an easy to use element, allowing an application to show a small pop-up for providing information for the user. It is mostly used to provide feedback about ongoing events, or to confirm that an action has been completed.

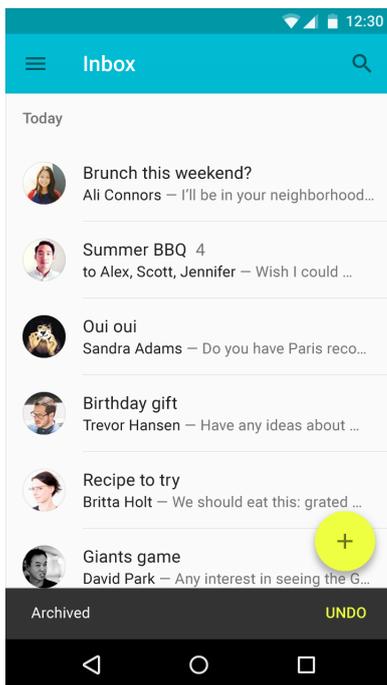
The Snackbar, shown in Figure 3.1c, also is used to show information, but takes up more space. It is mostly used to provide information about an action the user has made. Furthermore, it allows the user to react onto this, by pressing an included button. An example use case for this would be the deletion of an item, where the Snackbar could be used to inform about the successful deletion, but in the same time providing an undo action, via a dedicated button. [Cli15][Inca]



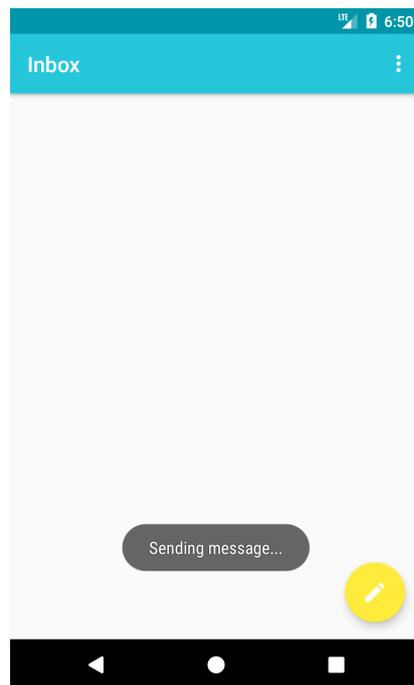
(a)



(b)



(c)



(d)

Figure 3.1: Material Design elements [Inca]

Methodology

For designing an application many different approaches can be taken. Three of the most elaborated and widely used are the methods User-Centered Design (UCD), Goal-Directed Design (GDD) and Activity-Centered Design (ACD). While each on its own has its strengths and weaknesses, the thesis will use UCD as its method of choice [Wil09]. This is because the proposed application should improve the way users engage with an ERP-System and also making their work more efficient, resulting in higher satisfaction. Therefore, since the focus is really on improving the users' experience, UCD fits best.

The following describes the idea of UCD and how it can be used, but also other methods used to develop this application.

4.1 Literature Research

At the beginning of this thesis, a comprehensive literature research was conducted. The goal was to lay the foundation for the later development of a mobile ERP-System. To be able to do this, first the idea of processes in such an application had to be researched. This led to a thorough investigation on sales processes, since this is the focus of this work.

Since the goal is to develop a functional prototype, including the connection to a database server, the technology needed to build such a system had to be researched

too. This includes the Android Framework but also network communication and synchronisation processes.

Therefore, online academic libraries, such as the ACM Digital Library [Liba] or the IEEE Xplore Digital Library [Libb], were used. This allowed to survey many different conference papers, articles, eBooks and journal entries, laying the theoretical foundation of this thesis. Most of these findings are presented in the Chapters 2 and 3, but also heavily influenced the work in the later chapters. For example the in Section 5.2 proposed synchronisation is based on different academic findings, to be able to implement a practical and elaborated process.

4.2 User-Centered Design

In addition to the general design rules, there are different approaches to the design process. The one that this thesis applies is called User-Centered Design. This approach heavily involves the user into the design process, as the name suggests. The next section presents the general aspects of an improved version of user-centered design, based on ISO 13407 and ISO 18529. This version of user-centered design consists of six processes and categorises them in two groups, usability engineering and user interaction design. This distinction is made to differentiate between the processes that are introduced through user-centered design and those, that are part of every design process. Furthermore, each process is specified through a precise description and the different outcomes each has [Jok02][134][185].

The first step of the user-centered design process is to define the user groups. This can be achieved through personas. Personas were first introduced by Allen Cooper in 1998. A persona is a model in the area of HCI (human computer interaction). It describes a fictional person, that represents a group of real users with certain behaviors and motivations. Furthermore, there is a differentiation into primary and secondary persona. The primary persona represents the group that the product is directly targeted at. The product is part of their daily usage. The secondary persona represents a group of users that is not paramount, but also uses the product. The result of this process is a precise definition of the target user groups for the product [Jok02][CLS08].

The second sub process is identifying the context of use. This includes the description of the users' characteristics, if the first process was not conducted with personas. Furthermore, the users' tasks and the technical, organisational and physical environment, where the product will be used, are analysed. As a result, the accomplishments of the user and the tasks to achieve them are described. Furthermore, the operational environment is defined, as well as the differences between the old and the new product [Jok02].

The third sub process deals with the user requirements. Its purpose is to define usability and interface interaction design requirements. Usability requirements comprise all aspects that have impact on the user experience (user effectiveness, efficiency and satisfaction), not only including the software, but also for example the user documentation. Following the general design guidelines and style guides is part of the interface interaction design requirements [Jok02].

The next sub process deals with user task design. Each relevant task that the user can perform with the system and the steps to accomplish each task, are thoroughly described in this process. The description for a task is uniformly structured. It always consists of a title that defines the action, like, for example, "create an order". This is followed by the scope that the task is part of, an actor, the intention behind the task and a precondition. The precondition defines where the user starts when performing the task. Lastly, a detailed description of the intended steps to accomplish the task is given. The result of this process is a precise definition of all relevant tasks [Jok02].

The next step is the usability evaluation. In this process, the current state of the product is evaluated against the context of use and the user requirements. Therefore, it is very important to include the user to achieve the most accurate results and suggestions for improvement. The reason behind this continuous testing is to reveal early on whether the usability of the evaluated product needs to be improved. Therefore, it is evaluated against five attributes, which are the learnability, efficiency, memorability, errors and satisfaction. When these criterias are met and fully satisfy the user requirements, the development can be continued. To be able to efficiently collect data from the users, different methods are available One approach is the thinking-aloud method, described in Section 4.3.3. This particular method has

been used during the design phase for this thesis application extensively. Further alternatives like questionnaires and interviews also provide very detailed information and can be used to deepen the analysis even more [Jok02][Sch01][CFH97].

The last sub process is the user interaction design. This process is the only one that is part of every design process. All other five sub processes are introduced through user-centered design and are part of the usability engineering category. The purpose of this process is to produce the user interaction and the visual design for the product. The process gets input from every other sub process. The challenging part is to integrate all the input into the design process. As previously stated, this process continuously develops the interaction and design over each iteration. Usually, the first iteration is just a drawing to communicate the initial baseline of requirements. Following this, clickable wireframes and mockups can be created to convey the look and feel of the system. This procedure allows to determine the user interaction design, before a single line of code is written [Jok02].

Except for the target user group definition, the user-centered design process is iterative. Though, there are many evaluation phases that lead to slight adaptations to the context of use and user requirements. Therefore, the steps to accomplish the user tasks can vary in each iteration, until no substantial improvements can be found in the evaluation process, anymore.

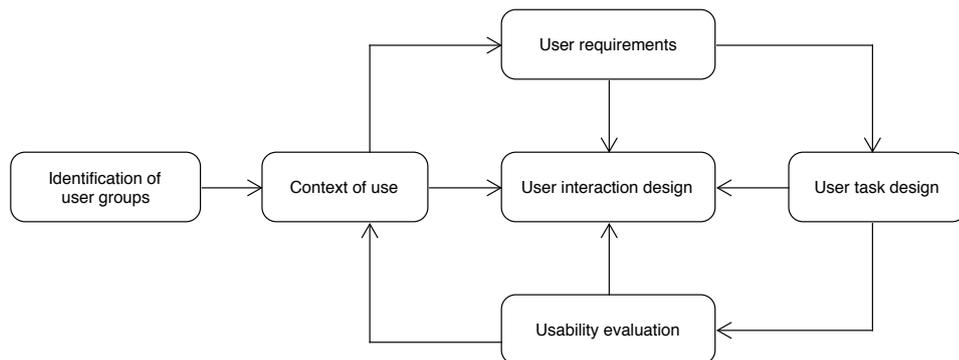


Figure 4.1: User-centered design process, based on [Jok02]

4.3 Evaluation Methods

In the following section, two different evaluation methods for HCI (Human computer interaction) and User Interface Design are presented. These will later be used to evaluate the mobile application, that will be developed over the course of this bachelor thesis.

4.3.1 Keystroke-Level Model

The first method is called KLM, which stands for “Keystroke-Level Model”. Its purpose is to determine, how long it takes to complete a certain task in a system. This provides a figure that allows comparison to other systems with the same task to measure its efficiency. It is an evaluation method from the GOMS family. Representing a simplified version of the original GOMS model (CMN-GOMS) that was introduced by Card, Moran and Newell in 1983. Generally, GOMS is an engineering model for interface design and is an acronym that stands for goals, operators, methods and selection rules. There are four different version of GOMS (CMN-GOMS, KLM, NGOMSL, CPM-GOMS) in use today, all based on the same GOMS concept [CMN86][JK96][SH10][Kie99].

In GOMS, the goals are the tasks the user wants to accomplish with this software. These tasks can be decomposed into unit tasks, that are smaller and cognitive more manageable [JK96].

The operators are a set of system and human actions to fulfil the users’ goals. This includes all the actions that are offered by the system like typing on a keyboard, selecting an item from a menu or button presses. Human actions are cognitive processes, for example, retrieving information from memory or preparing for the next step [SH10].

Methods are sequences of operators, which allow the user to reach a certain goal [CMN86][JK96][SH10].

Selection rules occur, when there is more than one method to accomplish the same goal or sub goal. The user chooses, based on his preferences, the method he is more efficient and comfortable with [SH10].

Since we are going to apply the Keystroke-Level Model, only keystroke-level

operators are used. This means only operators on a primitive level are used, like button presses or keystrokes, as well as a mental operator (a generalisation for all cognitive operations). Operators on a higher level of abstraction such as, previously stated, item selection from menus, are not used. KLM also excludes goals, methods and selection rules, which immensely reduces the modelling effort in comparison to the other versions of GOMS [JK96].

KLM usually consists of a set of six different operators, K (keystroking), P (pointing), H (homing), D (drawing), M (a mental operator) and R (the response time of the system). A keystroke is either the press of a key or a button on the mouse. The pointing operator describes the cursors' movement from point A to point B. Homing describes moving the hand from the mouse to the keyboard and vice versa. All these operators have a duration assigned to them. Standardised values for these durations were established through experimental studies, that can be used to achieve a fairly accurate result. The time expense of completing a task is simply the sum of durations of a sequence of these operators [CMN86][SH10].

Since this work evaluates a mobile application, the stated operators are not fully applicable. Therefore, an adapted version of KLM, that better fits the mobile touch interactions, like the FLM ("Fingerstroke-Level Model"), can be used. FLM reworked the naming, function and value of some KLM operators, leaving T (tapping – adapted from keystroking) and D (dragging – adapted from drawing) as new operators. P was partially changed. In the KLM, pointing was just the cursors' movement from point A to point B. In the FLM, the pointing operator is the fingers' movement from point A to point B with a tap inclusive at the target location. H only applies, when the hand of charge is changed. M and R retained the same. This allows for a more accurate analysis of execution times on mobile devices [Abd11][LSR⁺15].

4.3.2 Cognitive Walkthrough

The second evaluation method is the Cognitive Walkthrough. Its purpose is to evaluate, if the user interface, all its elements and their functions are clear to the user. It is an inspection method, that means, the evaluation is conducted by the

developer team and no test users are needed. Although it is conducted by the developer team, it uses the perspective of a novice user. This reflects the intuitive and explorative behavior of a new user [BvG02][Heg03][MSK10].

The first step in a cognitive walkthrough is the identification and prioritisation of all tasks that are relevant to the user. The next step is to identify the critical path for each task. This means defining the shortest and most likely path of actions the user will take to achieve a certain task. When the critical path for each task is defined, the experts now follow this critical path step by step. For each, evaluating the individual action according to four criteria [MSK10].

The first criteria is, if the user is trying to achieve the right effect. This means, trying to imagine if the user knows at each step, what to do next to get closer to the goal.

Following this, the experts evaluate, whether the UI-element for the next action is visible and perceivable.

Next, they assess if the connection between the UI-element and its effect is existent. They also evaluate, if other UI-elements may be considered instead of the intended one.

Lastly, the experts assess, if there is correct feedback from the system. This also includes, if the user performed the correct action, whether he is able to tell that he has made progress towards his goal.

Each of these criteria has to be answered for each tasks actions. Any uncertainties that occur, have to be analysed and rated by their severity. Furthermore, an appropriate solution for each has to be defined. These can range from small changes, like replacing an icon or changing its position, up to redesigning a whole mask [BvG02].

Cognitive walkthrough is a popular method for interface design evaluation. The reason for this- is, that it can be conducted at an early stage in the software development phase. Therefore, problems with the user interface can be identified and fixed early on, preventing high costs in the later stages of development [BvG02][Heg03][MSK10].

4.3.3 Think-aloud method

In the process of evaluating the prototype over the course of conducting the user-centred design process, the think-aloud method has been used to obtain feedback on the application.

The think-aloud method is based on the idea that the users, that participate in the evaluation, talk aloud while performing a task. It therefore generates direct data about what they think while trying to complete each task. With this, insights on possible struggles or problems with understanding how to go on can easily be identified. With this method it is important, that the observer only acts passively, which means that he does not respond to these articulated thoughts [JSvdBG04]. This way of using the think-aloud method is called concurrent think-aloud protocol. While this allows to observe the thoughts users have when using the application, it has to be noted that using this method influences the task performance negatively. It therefore should not be used to evaluate the efficiency of a system [vdHJS03].

The purpose of this method therefore is to better understand how a user perceives the application while using it to solve a task. Thus, it allows the observer to know how the user feels when the application is being used but also to get useful insights about the reasoning of decisions the users make.

Process Definition and Prototyping

5.1 Custom order creation process

For this bachelor thesis an adapted version of the general order creation process from Section 2.1.2 is created. This process takes advantage of the immediate access to the ERP system the field worker gains through the introduction of the mobile application. With this, it immensely reduces the, before regularly needed, communication between the field and in-house sales workers. Therefore, making the new process much more efficient and effective.

The new order creation process shares the same initial steps as the general creation process. The customer starts the process with a request. In-house workers then check if the master data for the customer already exists. If not, master data for the customer is created. After that, a meeting with the customer is arranged and the responsible field worker is informed by the in-house worker. At this point, the in-house worker is no longer involved in the order creation process.

The field worker now visits the customer at their desired location and timeframe. Together with the customer, the field worker checks the selected articles and informs the customer on new products, special offers and discounts on existing ones. The

big difference to the general order creation process in Section 2.1.2 is, that if any questions occur, the field worker can simply check the mobile application and does not need to consult the in-house worker for further information. The same applies, when the credit history of the customer is checked. Thus, the field worker now does not have to rely on the in-house worker for providing information anymore. This immensely reduces the time needed for creating an order. Furthermore, the field worker can immediately create the order entry through the mobile application. Thus, eliminating the redundancy of first noting it down on paper and later creating the order entry, as well as not involving the in-house worker anymore.

Although the adapted process has only slightly less steps than the general process, the reduction of time-consuming communication between the field and in-house workers results in a huge improvement. Consequentially, the customer satisfaction will increase, as well as the efficiency of the in-house workers, since they do not have to constantly provide information to field workers.

This adapted process can be seen in Figure 5.1.

5.2 The synchronisation process

As discussed in Section 2.3 the solution appears to be to use timestamps, to identify when the last synchronisation took place, combined with a two-tier replication scheme. Since two-tier replication supports mobile nodes and combines the benefits of an lazy-master-replication scheme and a local update scheme.

Therefore a process was developed to support this idea. This process consists out of three steps, which are shown in the Figures 5.3, 5.4 and 5.5.

Though, before looking at the specific processes the general style of communication will be presented.

Each synchronisation starts with establishing a connection with the server. Since both the mobile application and the database server have been written in Java, a TCP Connection via Javas default Sockets has been used for communication. It was decided to use TCP since this is a de facto standard for direct-connection network communication. In comparison, UDP would not be an option since it cannot be guaranteed that all data will be received, which would mean that complex error handling methods would have been needed. While with TCP you too cannot guarantee that the connection does not get disrupted, it supports the all or nothing principle. Since TCP makes sure all packets arrive and retransmits them if needed. In case of a network failure, it notifies the implementation by throwing an error [Com13].

Table 5.1: Available commands for the synchronisation

Command	Parameter 1	Parameter 2
Client sends		
!requestRecords	last synchronisation datetime	
!ack	article/customer/location/order/oa	received ids
!finished	sync	
!sendingOrders	number of records	
!sendingOas	number of records	
!finished		
Server sends		
!changedRecords	article/customer/location/order/oa	number of records
!changedRecords	done	
!ack	order/oa	number of records
!conflict	conflicting ids	

In this implementations case, the communication between the client (the mobile

application) and the server works by sending simple commands to request and announce the sending of data records. All used commands can be seen in Table 5.1. The messages that are exchanged are Base64 encoded to avoid third parties to easily spoof on the data. While the code was written to support encrypting the transmitted data too, it was not implemented, since this is not in the scope of this thesis. The architecture used to implement the communication is shown for the server in Figure 5.2a and for the client in Figure 5.2b.

Since the communication works the same for the client as for the server, in Figure 5.2b the communication is not modelled explicitly.

5.2.1 Step 1: Getting master data from the server

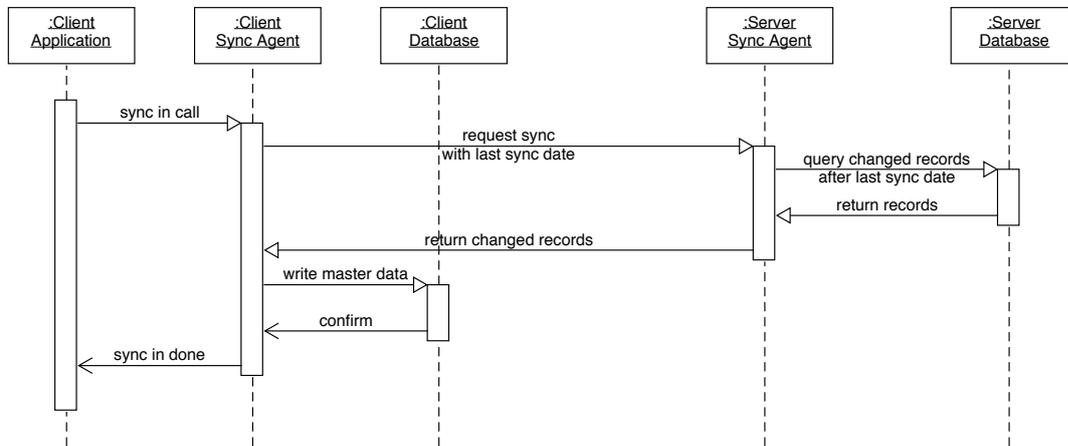


Figure 5.3: Step 1: Getting master data from the server

The first step, shown in Figure 5.3, starts with requesting all changes that occurred after the last synchronisation. Therefore, the client sends the command `!requestRecords` with the date and time the last synchronisation took place. The server receives this request and with its `SyncAgent` class then queries all changed records. It then parses the domain objects into JSON objects and sends them back, starting with the command `!changedRecords` and adding the data type and number of records to be expected. This is needed so that the client knows how to parse the message back into a domain object and also to be able to check if all records are received. Once all received data items are parsed, the client sends `!ack`, the data type and the received ids back to the server. With this the server is able to check if all records have been received. If there would be any missing, it would only resend the ones whose id has not been sent back. This process is repeated for each data type.

Once the server has sent all changed data records, it sends `!changedRecords done`, to inform the client that there are no more changed items. The client then confirms this by sending `!finished sync`. With this the first phase of the synchronisation is done.

5.2.2 Step 2: Checking for possible conflicts

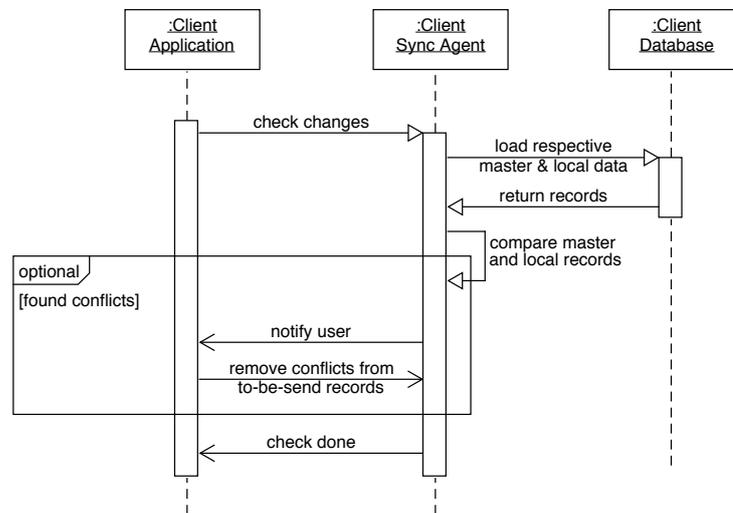


Figure 5.4: Step 2: Checking for possible conflicts

The second step, shown in Figure 5.4, is about finding possible conflicts between the local and just received master changes. It therefore loads all local changes and then checks whether the `id` is contained in the list of master records. If there are any matching `ids`, these will be stored and presented in the conflict resolution screen to the user. Here the user can decide which changes he wants to keep and which ones to discard. This works as described in the Section 2.3.1, where the interval synchronisation is described. For the time being the conflicting local records will not be sent to the server. The received master changes are written into the local master replication. And the local records will also be kept in the database. Once the user has resolved those conflicts, the local changes will be sent at the next synchronisation.

This step is done to reduce the amount of data that has to be sent over the network and also to avoid creating conflicts on the server side. Because if the client would just send all changes, depending on the interval the synchronisation takes places, it could be likely that there are already newer updates to those records. This would mean that the server would have to inform the client about those conflicts, leading to unnecessary communication between those two.

5.2.3 Step 3: Sending local changes to the server

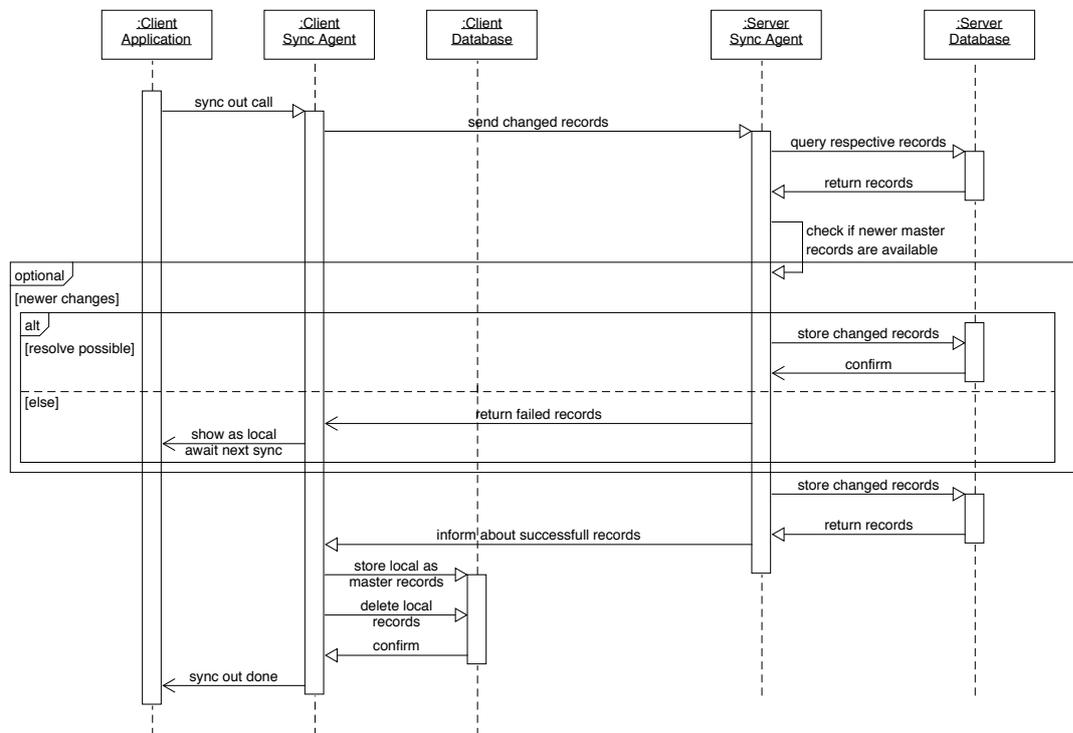


Figure 5.5: Step 3: Sending local changes to the server

The third and final step, shown in Figure 5.5, has as its goal to send all the local changes to the server. Therefore, it first notifies the server that it can expect to receive the local changes from the client. This is done by sending, depending on the data type that should be sent, either `!sendingOrders` for orders or `!sendingOas` for ordered articles. As parameter the number of records will be passed as well. After that, the data records will be parsed as JSON and are sent to the server.

The server now checks if there occurred any changes to these items in the meantime. While the timeframe is just really short, where changes could have occurred, depending on the network speed, it could happen that there are already newer changes. Since the master records just have been updated on the client and the client already checks for any conflicts before sending its local changes, this is only done to guarantee consistency. Which means making sure that no data, that has

not been resolved, will be overwritten.

If the server is able to resolve possible conflicts on its own, or if there are no such, the changes will be written into the master database. Any remaining conflicts, will be returned to the client, by sending `!conflicts` followed by the record ids. After that, all successfully updated records will be confirmed by sending `!ack`, either `order` or `oa` and the number of items. The client then checks if all items have been processed and then converts the local items to master items, deleting the matching local records in the database.

Finally the user is informed that the synchronisation has been completed. With this, the synchronisation process is finished and the connection will be closed.

5.3 Creation of the prototype

This section precisely describes the development process of the prototype for this thesis application using User-Centered Design, as described in Section 4.2. First, in Section 5.3.1, the target user groups for the application are defined using personas. Following this, the context of use is described in Section 5.3.2. Then, the user requirements are defined in Section 5.3.3. In Section 5.3.4, the relevant user tasks are precisely defined. The last section, Section 5.3.5, presents the different iterations of the user interface design process with each associated usability evaluation.

5.3.1 User identification

The focus of this bachelor thesis is the optimisation of the order creation by field workers. Thus, the main target group is the field sales force. The sales force consists of a broad spectrum of employees, including people of all ages, work experience and technological knowledge. When designing a mobile application, special attention has to be given to people that are not accustomed to using a modern mobile device. Therefore, this work focuses in the development phase especially on senior employees and novice mobile device users. Thus, meeting every user requirement for this group assures, that the entire field work force can utilise the application to its full extent.

The next user group is the sales administration. They are not paramount and make

up a way smaller group of users compared to the field sales workers. Nonetheless, they add additional user requirements that need to be taken into consideration in the development phase. These requirements regard the administrative work, like setting up and configuring the mobile application.

Throughout this prototyping process, cooperation and communication with a middle-sized sales business provides real world information. Furthermore, the access to the different user groups allows this work to very accurately evaluate the prototype from every user perspective.

The target user groups are further defined using personas.

Personas were first introduced by Allen Cooper in 1998. A persona is a model in HCI (human computer interaction). It describes a fictional person, that represents a group of real users with certain behaviors and motivations. There is a differentiation into primary, secondary and non-persona. The primary persona represents the group that the product is directly targeted at and developed for. The product is part of their daily usage and every user requirement is covered. The secondary persona represents a group of users that are not paramount, but also use the product. This user group usually adds product requirements, that are not covered by the primary persona, thus expanding the products scope. The non-persona typically describes the group of people, that are not targeted in the development phase. Mostly, these are very technical experienced people [CLS08][CRCN14]. For the mobile application, following persona were developed:

Primary persona

Frank H. is 43 years old. He lives with his wife and two kids in a middle sized single-family house close to the capital city. Frank is employed at a sales company. He works as a field worker in the sales workforce of the company. Franks income is somewhat above average. Next year he celebrates his 20th year anniversary at the company. He has a high school degree, but never went to college. Though, he has completed a few seminars and workshops for sales strategies as further education.



His personality is best described as extrovert, cooperative, traditional, friendly and a bit anxious towards dealing with new technology. He is more of an old-school type of person, struggling to keep up with his kids' growing affection towards smartphones and computers. He recently got a smartphone as a present from his wife at his 43rd birthday, barely using it, since he does not want to concern him with this in his leisure time.

Secondary persona

Jessika K. is 37 years old. Jessika is divorced and lives with her two children in a mediocre apartment. She used to work as an in-house sales worker for a little over a decade. Just recently, a few months ago, she got promoted to junior ERP administrator. Jessika is not familiar with the whole ERP system, since she was only frequently using a small part of it in the sales process. Jessika is still struggling, since managing the whole ERP system is overwhelming for her. Although Jessika owns a smartphone, she limits the use of it to the most basic applications.



Jessika is a very friendly, understanding, a little bit clumsy but very committed person. Once she sets a goal for herself, she sticks to it until it is achieved.

Non-persona

Gerald D. is 31 years old. He shares his apartment with his current girlfriend. Gerald completed his study in informatics at college successfully. He is currently the ERP administrator of a sales company. He is very interested in technology. Naturally, this is demanded by his job, but he also spends a lot of his leisure time keeping himself updated on the latest technological advances. He owns various kinds of hardware, like the newest flagship Android phone, a laptop, a desktop-pc, a tablet and uses them daily.



His close friends describe him as curious and eager to learn new things, introvert, patient and understanding. After work he likes to meet up with friends, or just enjoy a lazy evening with his girlfriend and the newest series on Netflix.

5.3.2 Context of use

This section presents the context in which the mobile application will be used. The context of use describes the users, their tasks and the technical, organisational and physical environment. A more detailed definition of the user groups is found in Section 5.3.1.

The main target users of this application are the field sales workers. The most relevant task for the field worker is creating orders, as described in Section 5.3.4. The technical environment for this application are mobile devices. Thus, there are various restricting factors, such as a smaller screen size and storage, the input method via touch and the network reliability.

The organisational environment specifies which user has access to which parts of the application. Furthermore, the available information for each user can be restricted. The field worker needs access to article and customer master data, as well as the authorisation to create order entries. Generally, the topic of security is very delicate, regarding access to information from outside the business and data transmission over the internet. Therefore, the devices must use appropriate authorisation measurements and the transmitted data should be encrypted.

The application is used anywhere on the way to and at the customer. Thus, external environmental factors such as weather and lighting need to be considered. Moreover, the limitations of interacting with the mobile device while moving should also be accounted for. These two form the physical environment of the application.

The secondary target user group are the sales administrators. Their main task is to set up and configure the application. Their physical environment is mostly restricted to indoors, since administrative work is done in an office and not at the customer. Since they have to configure all parts of the application, their organisational environment is not constricted, in terms of information accessibility. They share the same technical environment as the sales workers.

5.3.3 User requirements

This section describes the user requirements for the application, representing the users' needs towards the product. These need to be considered in the development process to provide a satisfying user experience. User experience is a broad term, that describes the experiences a user has with a product, regarding performance, effectiveness and satisfaction [JG07].

This thesis conducted interviews with employees of the cooperating sales business, to define user requirements [MB02].

First, the important aspects of the applications' performance haven been discussed. This included immediate availability and short loading times. Thus, this aspect was crucial in the development of the database structure and synchronisation process, as described in Section 5.2.

Secondly, a user requirement regarding efficiency has been, that the order creation workflow does not require a lot of keyboard typing. As a result, we implemented the order creation process to automatically load as much information as possible from the master data, as precisely described in Section 5.3.4.

Regarding satisfaction, the users requested that the user interface is well-structured and not overcrowded. This helps to quickly locate information. This requirement is met by strictly following the Android and Material Design guidelines and has been put into practice throughout the application.

A further requirement has been, that the application has the look and feel of a typical Android application. This means, that standard user interface elements are used and the positioning and alignment of the elements adhere to the Android design guidelines. Moreover, the functionality of the user elements should be according to the users' expectations, for example that the back-button brings the user one step back. This bachelor thesis met this requirement by strictly following the Android and Material Design guidelines.

5.3.4 User tasks

Title	Create Order
Scope	Managing Orders
Intention	User wants to create an order
Precondition	User has to be in the order view, or in the related article\customer view
Main Scenario	<p>The user first locates the FAB (floating action button) and clicks it. This opens the order creation view. A popup appears, where all the available customers are listed by name. The user either types the name of the desired customer in the dedicated search field, or scrolls through the list until the name appears. Following this, the user selects the customer, which opens a new popup to choose the location of the customer. The same procedure as in the customer selection is applied. When the customer location is selected, the popup closes and the customer data is automatically loaded into the designated text-fields.</p> <p>Description of the text-fields:</p> <ul style="list-style-type: none"> • Order no.: Automatically filled and cannot be changed • Order date: Automatically filled, but can be changed by clicking on the date itself or the triangle on the right side of the date. This opens a date-picker to choose the desired date. • Shipping date: Automatically filled with a default of 10 days after the order date, can be changed with the same procedure as in order date. • Customer: Automatically filled with the customer name of the selected customer. • Customer location: Automatically filled with the name of the selected customer location. • Contact name: Automatically filled with the name of the contact of the selected customer location. • Shipping address: Automatically filled with the deposited address of the customer location, but can be changed. To change the shipping address, the user selects the field and deletes or types in new information. • Billing Address: Automatically filled with the deposited address of the customer location, but can be changed with the same procedure as in shipping address. • Shipping method: Initially empty. To add a shipping method, select the field and type in a shipping method. Frequently used methods are suggested by the system and can be selected. • Payment method: Initially empty. To add a payment method, the same procedure as in shipping method is applied. <p>After the user has entered all information, the next step is to either locate the “ADD ORDERPOSITIONS” button and click it, or swipe to the right. This opens a view, where order positions can be added to the order. The user can search for an article by typing the name in the search-field, or scrolling through the list of articles. The user selects the desired article by clicking on its name. The article appears as an order position in the right area.</p>

5. PROCESS DEFINITION AND PROTOTYPING

	<p>Description of the text-fields:</p> <ul style="list-style-type: none"> • Item no.: Automatically filled with the article ID and cannot be changed. • Item description: Automatically filled with the deposited description of the article and cannot be changed. • Quantity: Automatically filled with the default value of 1 and can be changed by selecting the field and typing in the desired quantity. • Unit: Automatically filled with the default value of PCS and can be changed by selecting the field and choosing the desired unit from a dropdown menu. • Discount: Automatically filled with the default value of 0.00 and can be changed by selecting the field and typing the desired discount (values in percent from 0.00 - 100.00) • Price: Automatically filled with the price of the article, cannot be changed. <p>The user can delete an order position by clicking the associated delete-icon. When all order positions are added, the user locates the checkmark icon and clicks it. Following this, the order view opens and the newly created order is selected. The order creation process thus is complete.</p>
--	---

Title	Edit/Delete Order
Scope	Managing Orders
Intention	User wants to edit or delete an existing order
Precondition	User must be in the order view. The order must be either in the state "PROCESSING" or "IN PROGRESS".
Main Scenario	<p>The user first selects the order he wants to edit or delete. When the desired order is selected, the user locates the edit-icon and clicks it. This opens the order edit view. This view has the exact same text-fields as the order creation view, automatically filled with the current information of the order. All information can be changed as explained in the order creation process. The order positions can be added or deleted, and the values for each changed with the same procedure as in the order creation process.</p> <p>When all desired changes are applied, the user accepts the changes by clicking the checkmark-icon. This opens the order view and the changed order is selected.</p> <p>The user is able to delete the order by clicking the trash bin icon in the edit view.</p>

5.3. Creation of the prototype

Title	Search and filter
Scope	Managing Orders
Intention	User wants to search and filter orders
Precondition	User has to be in the order view
Main Scenario	<p>The user first locates the search-area. To filter the search-list, the user has to locate the filter icon and click it. This opens a dropdown menu with four menu-items. Following this, clicking the first menu item, “Filter state”, opens a submenu. This submenu contains three menu-items, “Processing”, “In progress” and “Finished”. All the menu items are selected by default. By deselecting, for example, “Processing”, the search-list now refreshes and shows only orders that have the state of “In progress” or “Finished”. This procedure can be repeated in any combination.</p> <p>The search-list by default is sorted in descending order by order number. This can be changed by clicking on the filter-icon. This opens the dropdown menu.</p> <p>Description of the menu items:</p> <ul style="list-style-type: none"> • Filter state: opens submenu with filter options as described before • Order no.: This item is by default selected. When this menu-item is selected, the search-list is sorted in descending order by order number. If the user clicks the search-box and queries for an order, the search-list only shows orders, where order number contains the query. • Order date: When this menu-item is selected, the search-list is sorted by order date from newest to oldest. If the user clicks the search-box and queries for an order, the search-list only shows orders, where the order date contains the query. • Customer name: When this menu-item is selected, the search-list is sorted by the customers’ name of the order from A to Z. If the user clicks the search-box and queries for an order, the search-list only shows orders, where the customer name contains the query.

Title	Conflict resolution
Scope	Managing Orders
Intention	User wants to resolute conflicting orders
Precondition	User has to be in the overview screen and there are conflicts
Main Scenario	The user locates the conflict resolution card in the overview screen. By clicking the “RESOLVE NOW” button, it opens the conflict resolution view. The user can choose between three different resolution strategies.

	<p>Description of resolution strategies:</p> <ul style="list-style-type: none"> • Master data first: When this strategy is selected, the system accepts the master data and discards the local data. By clicking the “RESOLVE NOW” button, the master data is accepted and the conflict resolution is complete. • Local data first: When this strategy is selected, the system accepts the local data and discards the master data. By clicking the “RESOLVE NOW” button, the local data is accepted and the conflict resolution is complete. • Manual resolution: When this strategy is selected, the user can choose from both, the local and the master data, which data for each conflicting field should be accepted. The user can select only either the value of the master data, or the local data for each field. The user must select one value for each conflicting field. Afterwards, by clicking the “RESOLVE NOW” button, the selected data is accepted and the conflict resolution is complete.
--	--

5.3.5 Prototype creation

During this process multiple evaluations have been conducted. Starting at the wireframe and at all following phases a questionnaire has been used too, to get a deeper understanding of the users’ feedback. Furthermore, this allowed to compare the progress made during each step. This questionnaire included both general and navigation related topics, where each question could be rated with 1 to 5 (1 is worst and 5 is best). Moreover, the participating users could also rate the experience solving the use cases with the same rating schema.

Sketch

The first step in the development of a prototype is the sketch. The purpose of this is to give a rough overview on the possible structure of the application. Furthermore, the later needed elements can be specified too. For example, the Figures 5.6 and 5.7 show the sketches of the overview and article screen. Various base android elements are used, such as the navigation bar and the navigation drawer, as well as the floating action button in the overview screen. For the content, a two-pane layout was chosen to most efficiently use the space, available on the tablet. On the left side, there is a designated search area with a search-box on top. The right side provides either general information and statistics, as seen in Figure 5.6, or detailed information of an article, shown in Figure 5.7.

The evaluation of the sketch showed, that it was very difficult for novice users to perceive the navigation. Therefore, for the next revision, permanent icons have been added to the layout in form of a small, persistent version of the navigation drawer. These icons are more apparent and can be used to switch through views as an alternative to the normal navigation drawer.

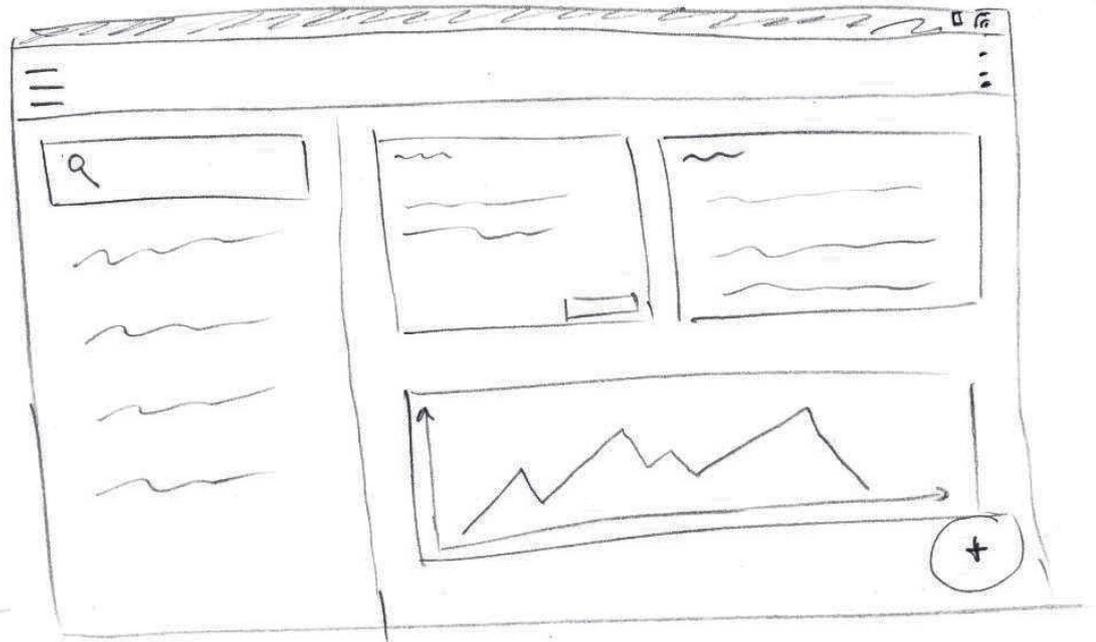


Figure 5.6: Sketch of the overview screen

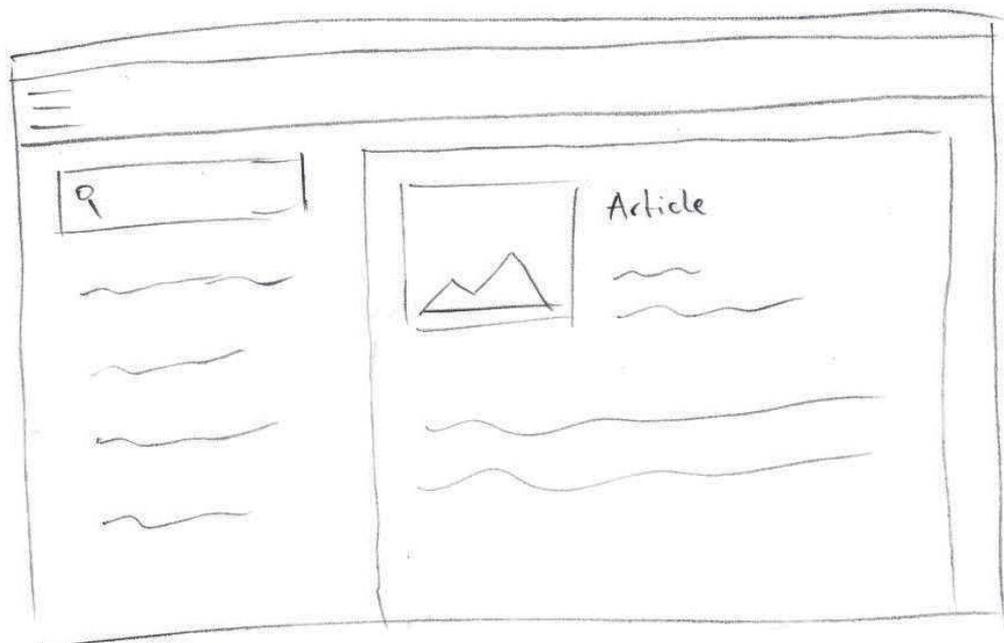


Figure 5.7: Sketch of the article screen

Clickable wireframe

The next step is to create a clickable wireframe. The purpose of this is, to model the interface interaction and possible workflows with a more detailed design and more realistic data. The sketches provide the basis for the structure and layout of the wireframe and the insights from the evaluation are adopted. The small navigation drawer is implemented on the left side of the content layout, the floating action button is repositioned to the order view and generally more details are added, shown in Figures 5.10 to 5.13. In the design process of this wireframe, the general Android and Material Design guidelines are already considered. Thus, only stock Material Design icons are used. Moreover, the placement, spacing and alignment of interface elements are inspired by the style guides.

The most important aspect of this wireframe is the modeling of the order creation process. The workflow for the order creation is shown in Figures 5.12 and 5.13. Since the available space on a mobile device is significantly less than in usual ERP systems, the first approach is to split up the order creation process. First, the general order information is entered and in the following steps the articles are added as order positions. The idea is to add one article at a time, as shown in Figure 5.13.

The wireframes' evaluation revealed many weaknesses of the order creation workflow. First, the sequential addition of articles to the order is way too inefficient, especially when adding a large quantity of articles. Secondly, keeping track of what articles have already been added and generally navigating through the chosen articles is very tedious and inefficient. Furthermore, reworking the overview screen has been suggested. The search-area in the overview screen is unnecessary and should be replaced with more important general information. The evaluation also showed, that some of the chosen navigation icons are not optimal. Most test users did not recognise the order and customer-icon. Thus, the icons need to be reworked as well.

Finally, for the next prototyping step, real data has been provided by the company, with which the evaluation has been conducted.

5. PROCESS DEFINITION AND PROTOTYPING

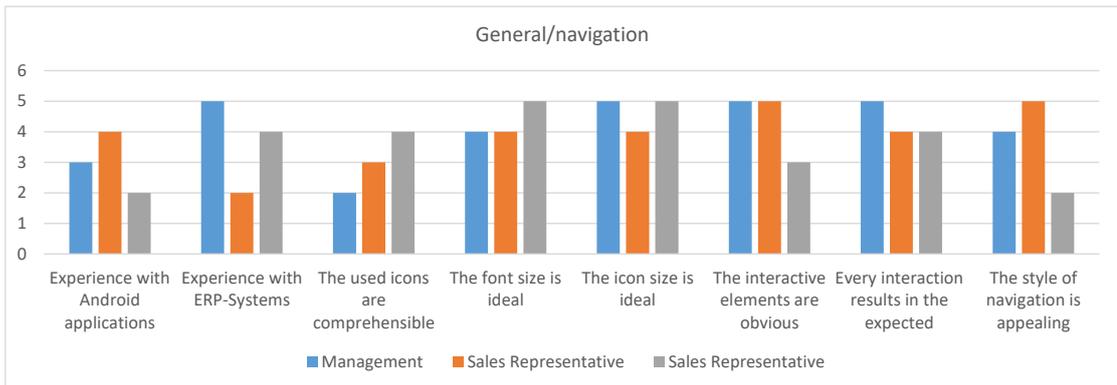


Figure 5.8: The results for the general and navigation related questions

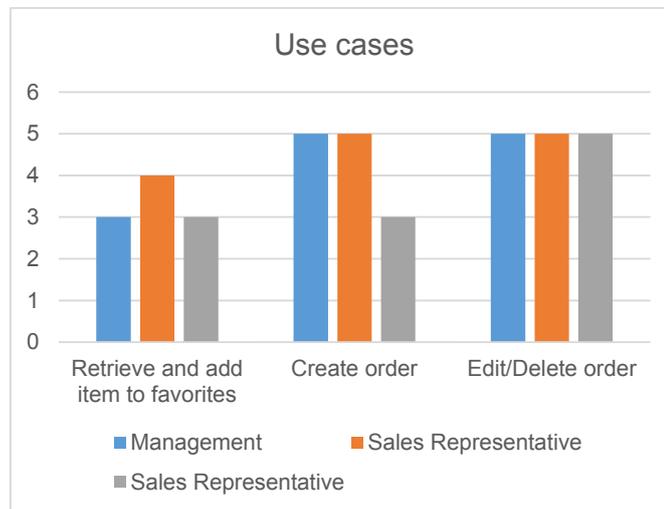


Figure 5.9: The use case rating

5.3. Creation of the prototype

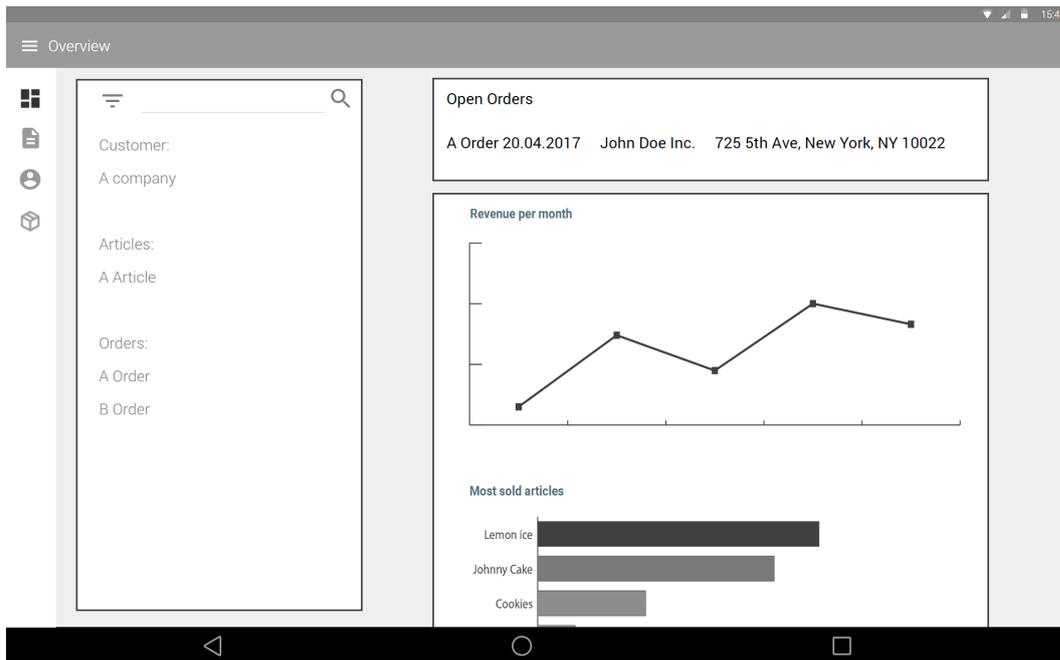


Figure 5.10: Overview screen of the wireframe

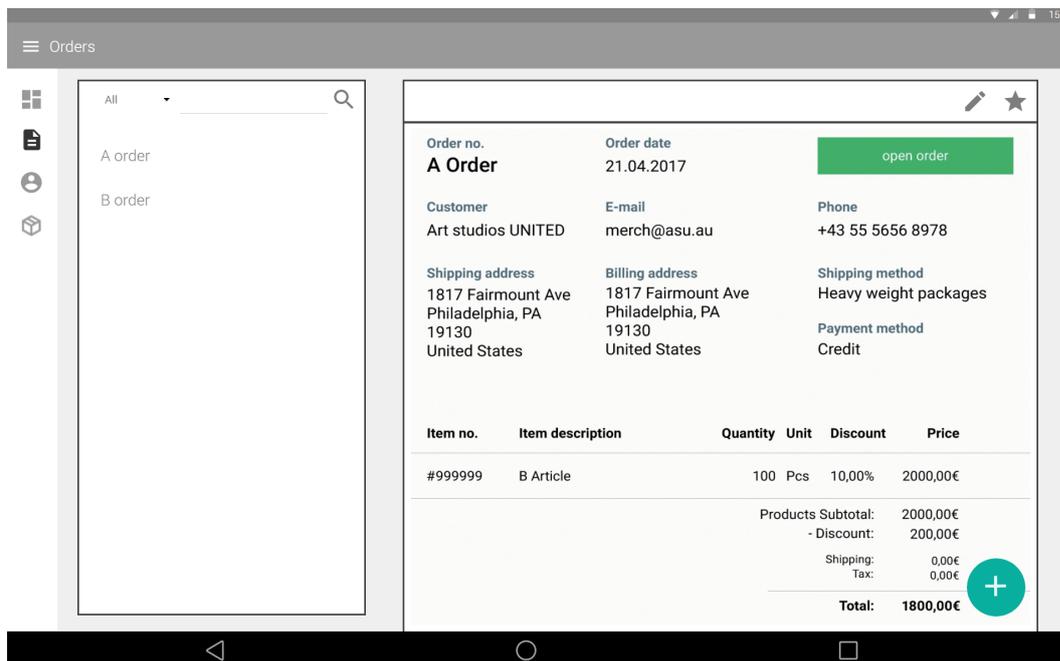


Figure 5.11: Order screen of the wireframe

5. PROCESS DEFINITION AND PROTOTYPING

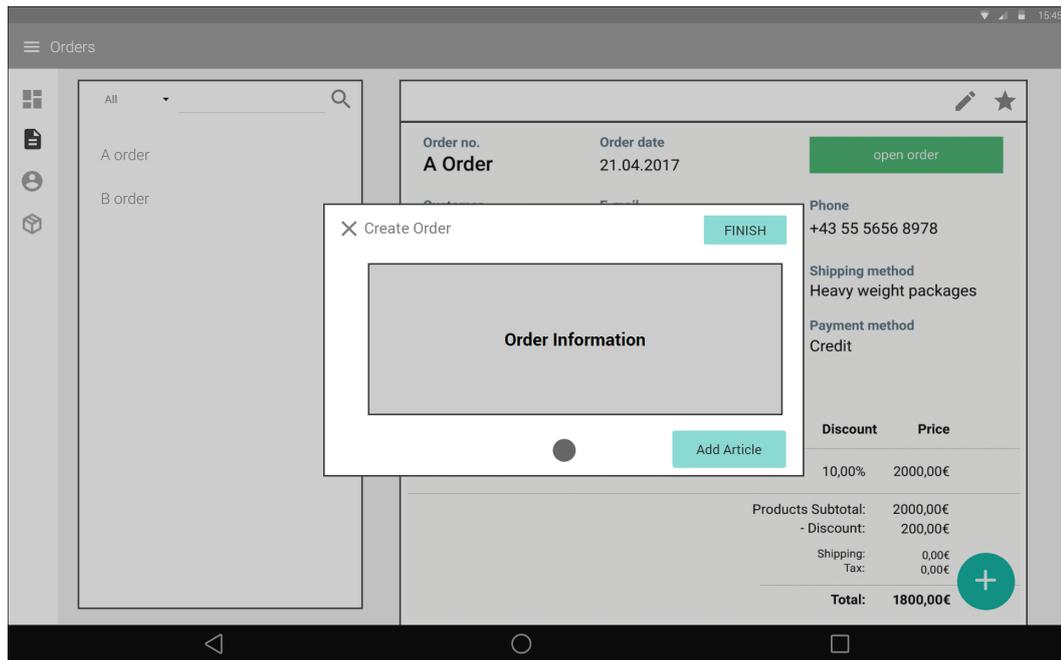


Figure 5.12: Order creation process of the wireframe (part 1)

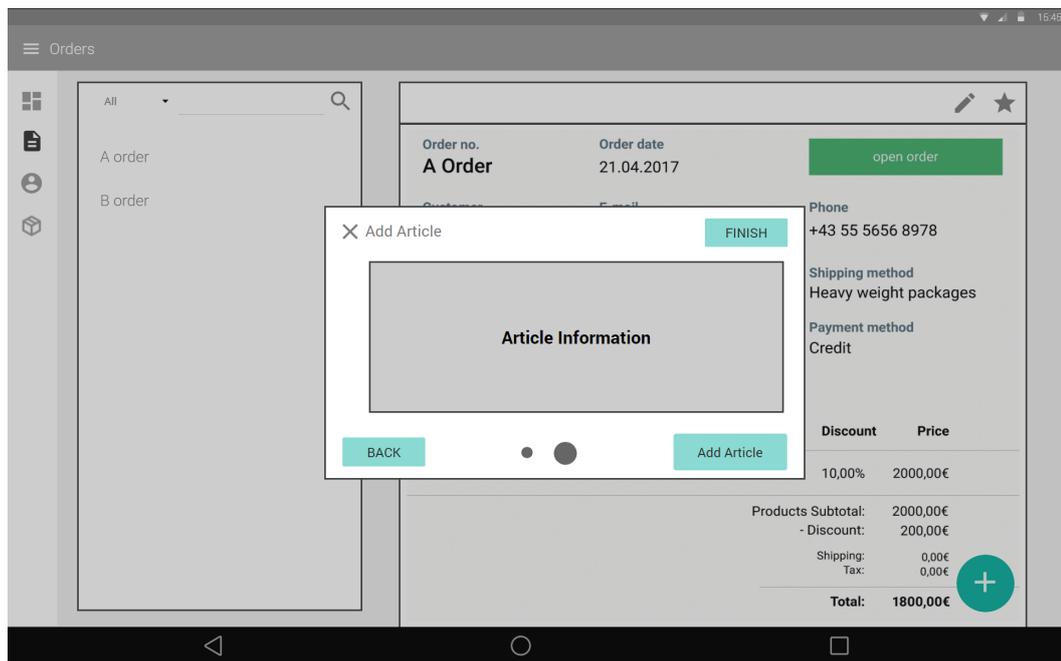


Figure 5.13: Order creation process of the wireframe (part 2)

Clickable mockup

The next step in the development of the prototype is creating a clickable mockup. The purpose of this is to develop the final design for the prototype.

The wireframe and the suggestions from its evaluation are the basis for the design of the mockup. In the design process of this mockup, the general Android and Material Design guidelines are precisely followed. Thus, for the spacing, placement and alignment, the standardised values, that Material Design defines, are used. Furthermore, all interface elements that are used, are provided by Android. Therefore, it is guaranteed, that all elements, that are used for the design, can be implemented in the prototype. Moreover, a color scheme was added. This scheme was created with the Material Design Color Tool [Too]. The usage of color is kept simple and highlighted elements are colored in a harmonising accent color. In addition to that, elevation was added to interface elements, to visualise contrast and hierarchy. This is a main guideline of Material Design.

The most substantial change in the mockup is the rework of the order creation workflow, since its inefficiency was criticised heavily in the wireframe evaluation. The general idea of splitting the order creation in two parts remained. However, the popups that were used in the wireframe are replaced with entire views designated to the process, shown in Figures 5.18 and 5.19. First, this approach provides more space for displaying information, which improves the readability and helps the general structures' clarity. Secondly, the process of adding articles to the order was completely revamped. The application displays all articles in the designated search-area, that provides the same search functionality as in the general article screen. By selecting articles in the search-area, a Snackbar (a Material Design element, shown in Figure 5.19.) appears, that displays the number of selected articles. By clicking the "ADD TO ORDER" button, the articles are added to the order as order positions. This process allows for a faster article selection and the right side, as shown in Figure 5.19, always displays the current order positions. Compared to the wireframe, keeping track of what articles have already been added to the order is immensely simplified. Aside from the reworked order creation workflow, the other suggestions from the wireframe evaluation were adopted. Thus, the search-area in the overview screen was replaced with general information and

5. PROCESS DEFINITION AND PROTOTYPING

the order- and customer-icons were replaced.

The evaluation of the mockup showed, that changes of the order creation workflow, made in the mockup, are an immense improvement and can be adopted in the prototype. Furthermore, the icons were perceived as more intuitive. Aside from that, the only suggestion for improvement was to increase the text-size, to provide better readability.

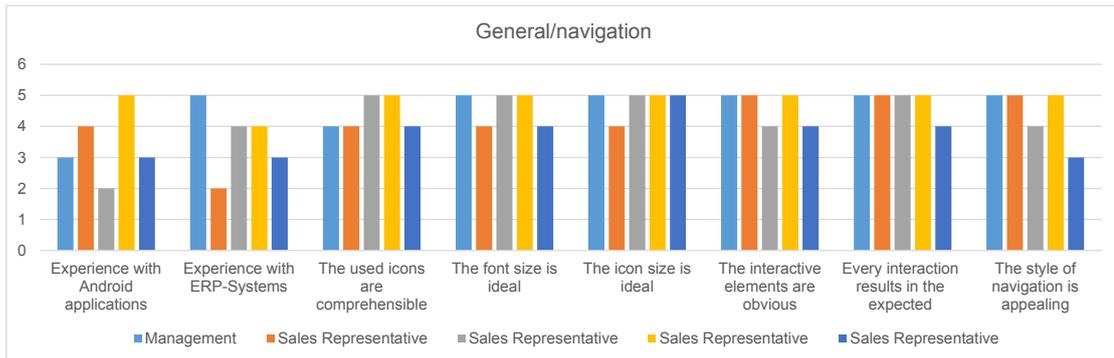


Figure 5.14: The results for the general and navigation related questions

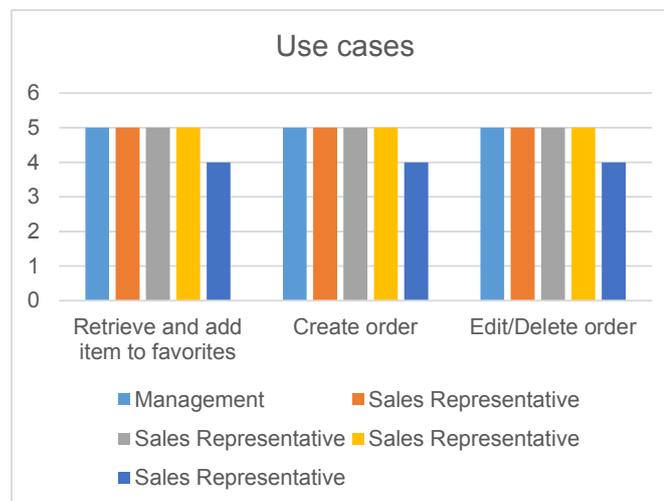


Figure 5.15: The use case rating

5.3. Creation of the prototype

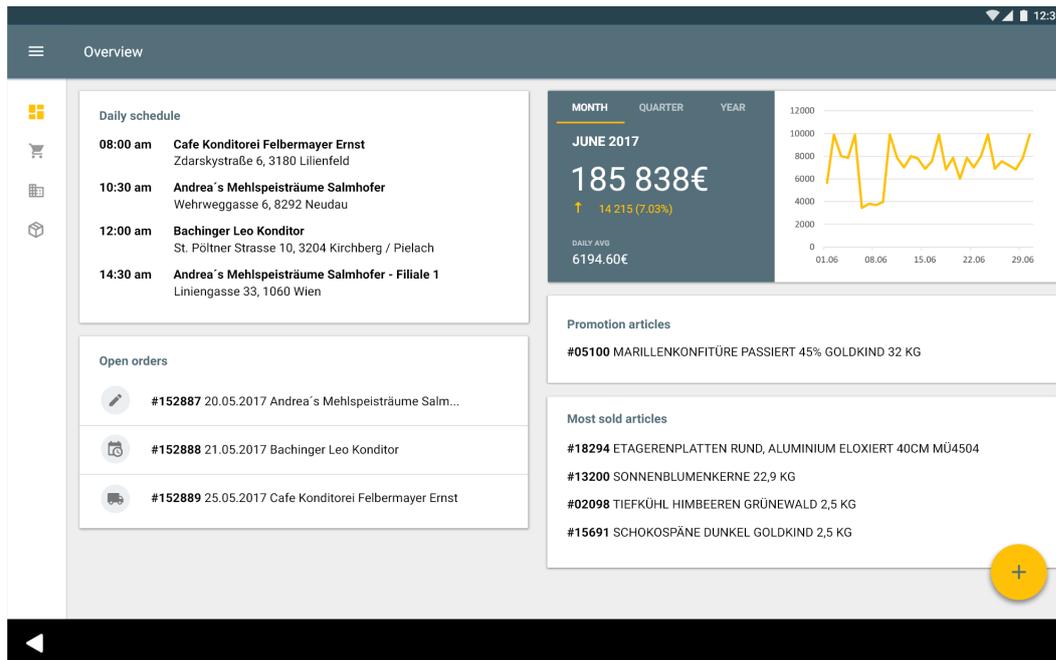


Figure 5.16: Overview screen of the mockup

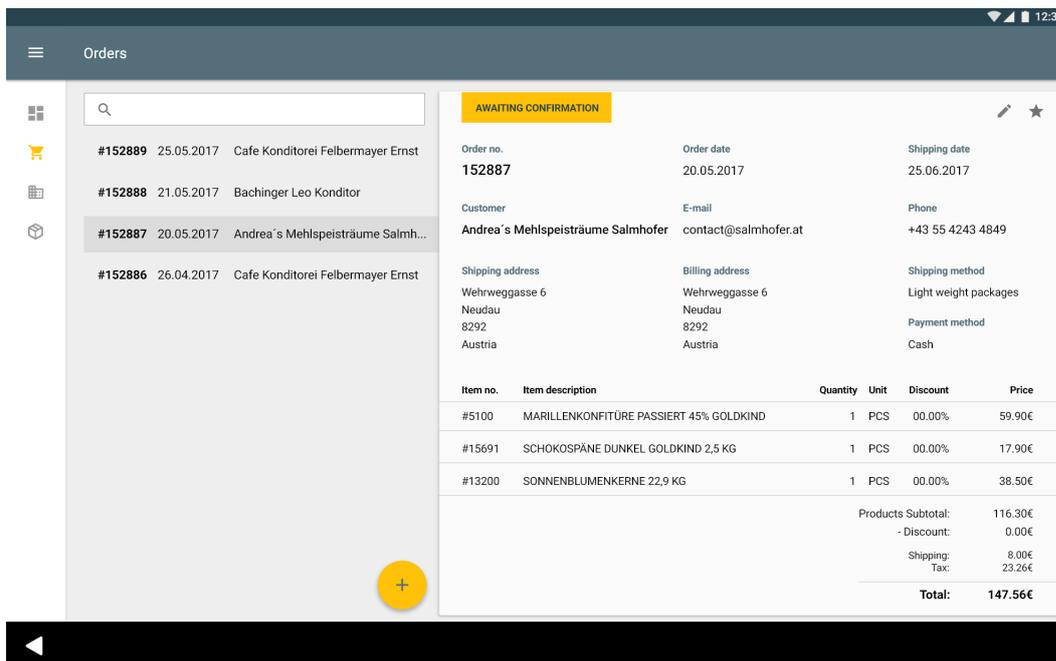


Figure 5.17: Order screen of the mockup

5. PROCESS DEFINITION AND PROTOTYPING

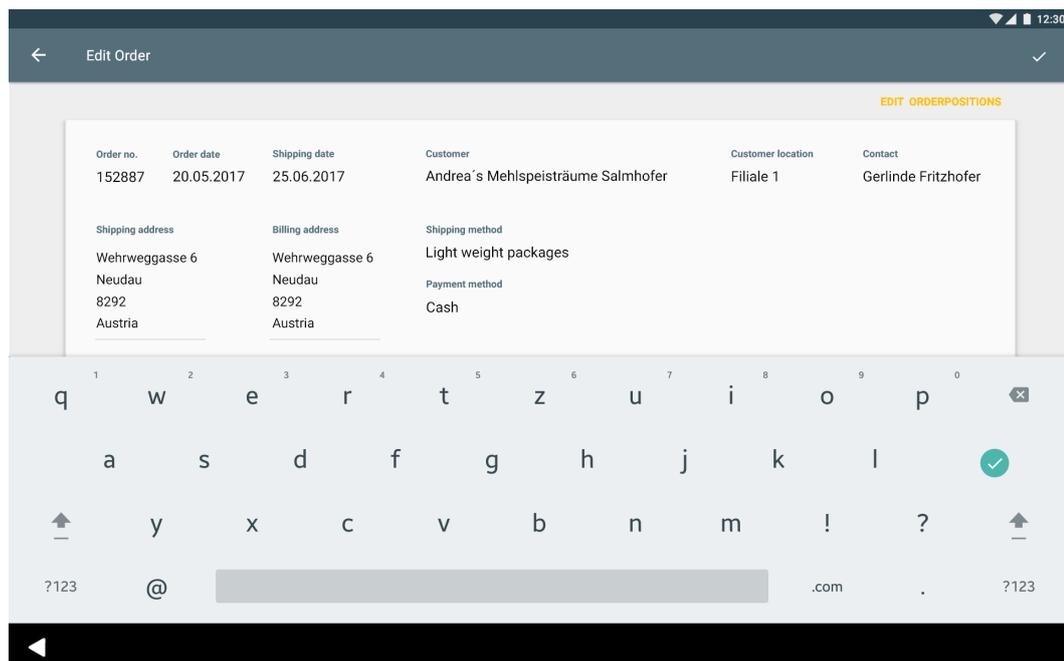


Figure 5.18: Order creation process of the mockup (part 1)

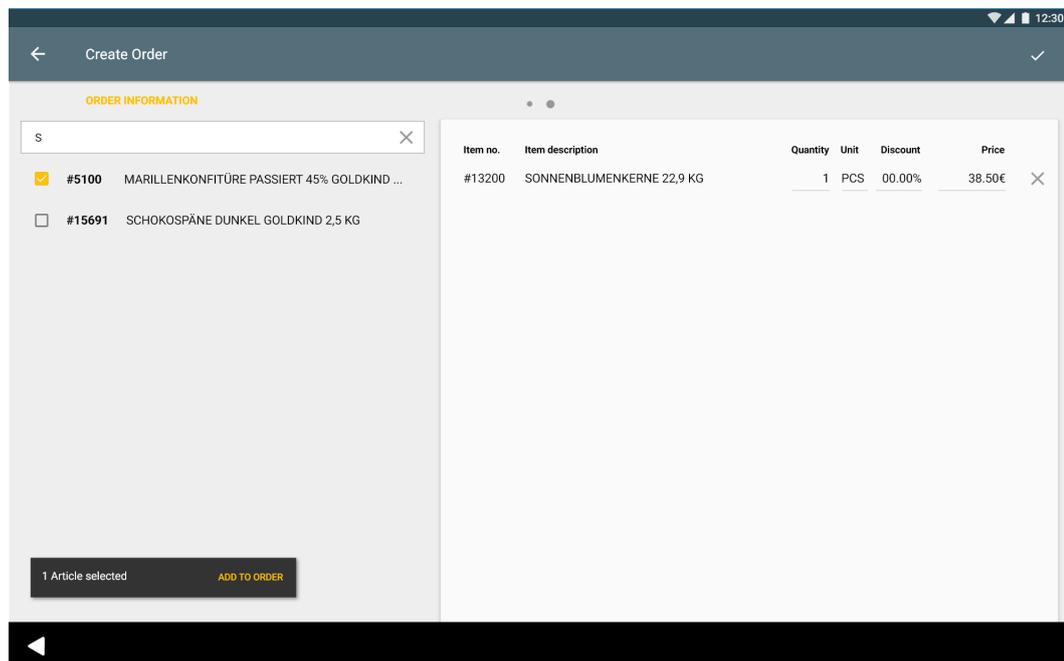


Figure 5.19: Order creation process of the mockup (part 2)

Prototype

The last step in the development of the prototype is creating the actual prototype. The mockup and its evaluation lay the foundation for the implementation. The design and interaction of the mockup, that was thoroughly evaluated in the previous development phases, is now implemented. Since the clickable mockup already followed the Android and Material Design guidelines, only slight adjustments had to be made. The Android framework was used to implement the prototype.

The prototypes' evaluation results were very positive. The test users criticised nothing in the overview, customer and article screen. Only small details in the other screens are mentioned. For example, the search-area in the order screen should display the order-state through either a color accent or a small text addition. Furthermore, the users did not recognise that the customer name in the general order information is clickable. This should be made more apparent through an indicator. Moreover, the icon for the order view in the small navigation drawer is still not optimal. Lastly, a small detail in the order creation process was criticised. Some test users mentioned, that the multi-selection of articles is not very intuitive. Instead, the article should immediately be added as order position just by selecting it.

Following this, the suggestions from the evaluation were implemented. Generally, interface elements that can be interacted with, are made more apparent through indicators. Application settings were implemented, that allow to enable or disable the multi-selection in the order creation process and state indicating colors for the order search-area.

These final changes are already adapted in the Figures 5.22 to 5.25. Thus, these show the final version of the prototype.

5. PROCESS DEFINITION AND PROTOTYPING

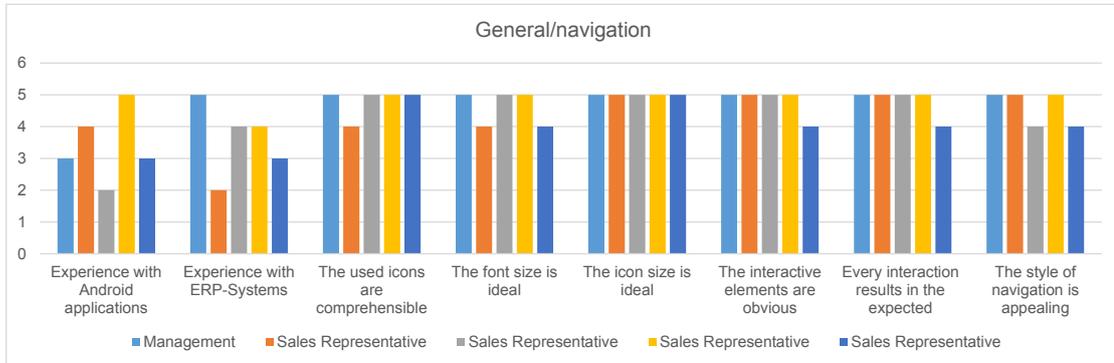


Figure 5.20: The results for the general and navigation related questions

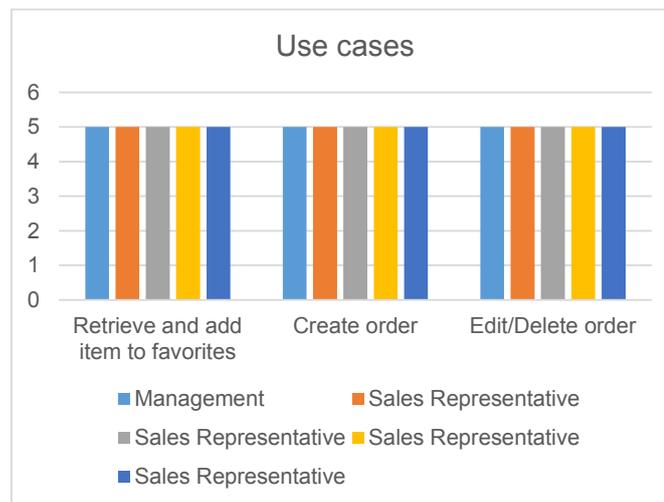


Figure 5.21: The use case rating

5.3. Creation of the prototype

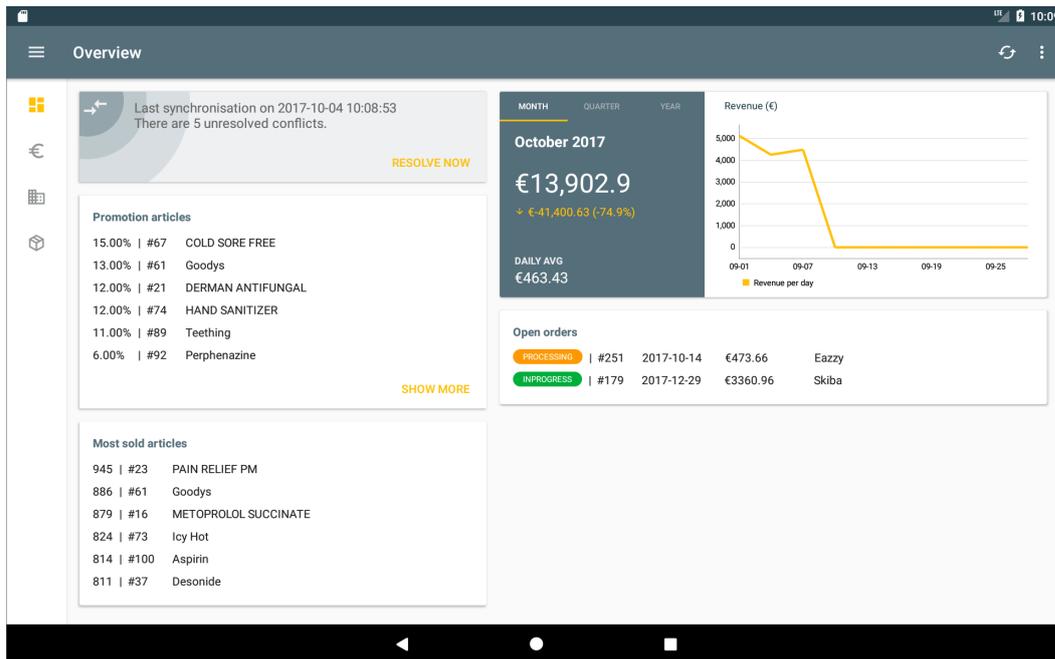


Figure 5.22: Overview screen of the prototype

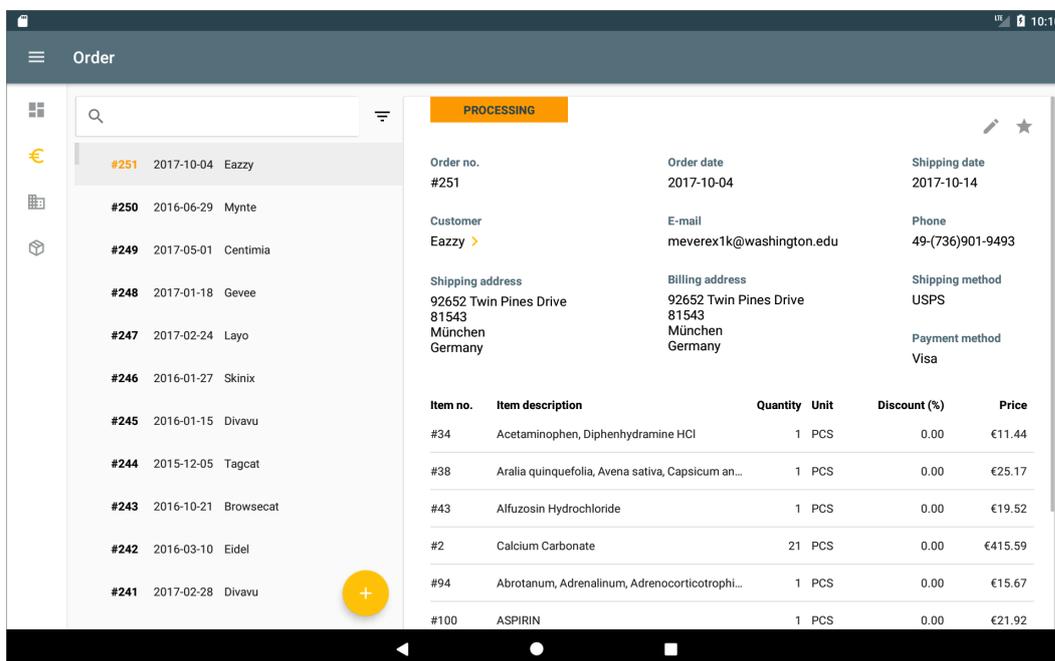


Figure 5.23: Order screen of the prototype

5. PROCESS DEFINITION AND PROTOTYPING

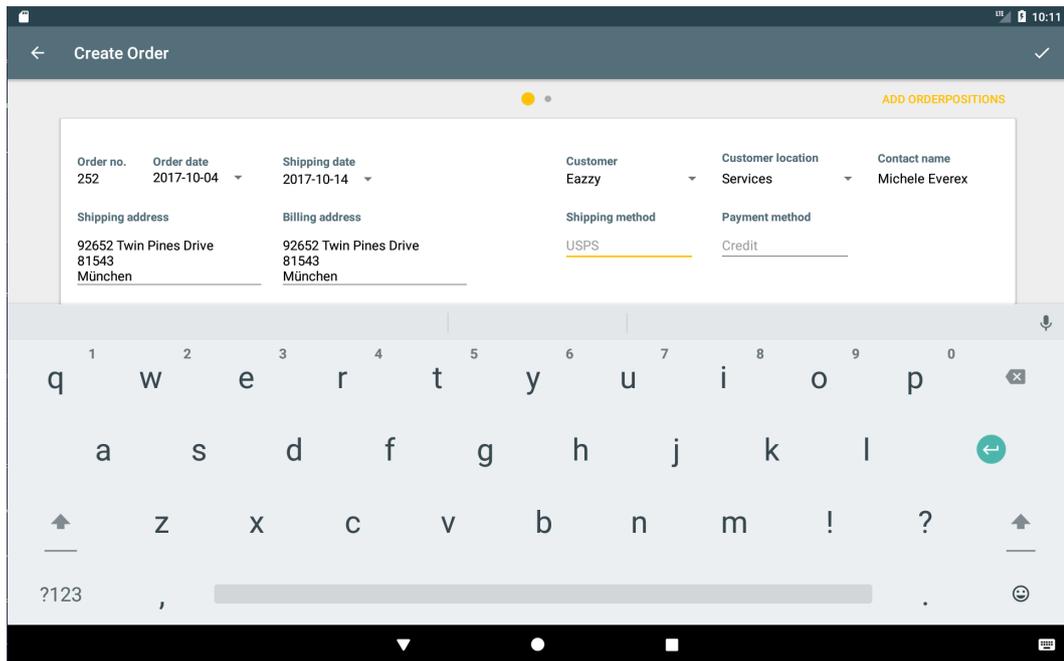


Figure 5.24: Order creation process of the prototype (part 1)

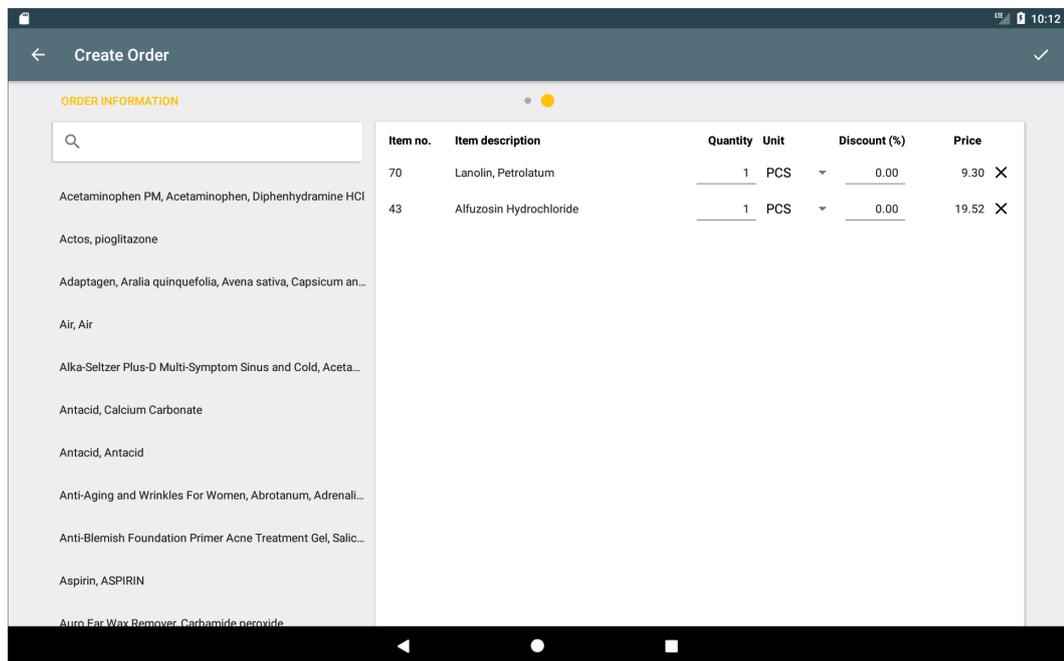


Figure 5.25: Order creation process of the prototype (part 2)

Evaluation

6.1 KLM

The following section evaluates the efficiency of the order creation process, that was developed over the course of this bachelor thesis. For this, the adapted version of the KLM (Keystroke-level Model), the Fingerstroke-level Model, as described in Section 4.3.1, is used to measure the duration of the whole order creation process. Following this, the order creation process of another ERP system (a desktop version) was analysed with the general KLM to provide a comparison.

After the order creation process in each system is precisely analysed, the results will be listed and compared in Section 6.1.3.

6.1.1 Fingerstroke-Level Model

In this section, the Fingerstroke-Level Model is applied to the application, that was developed over the course of this bachelor thesis. The operators used are T (for tapping) with the duration of 0.31 seconds, P (for pointing) with the duration of 0.44 seconds, M (for mental operator) with the duration of 1.35 seconds and lastly, R (for response time of the system) with the duration of 0.35 seconds. The values for the operators T and P are established in the the study of Lee [LSR⁺15]. The mental operator M has the standardised value defined by the original creators of KLM [CMN86]. The system operator R is variable, dependent on the response time of the application.

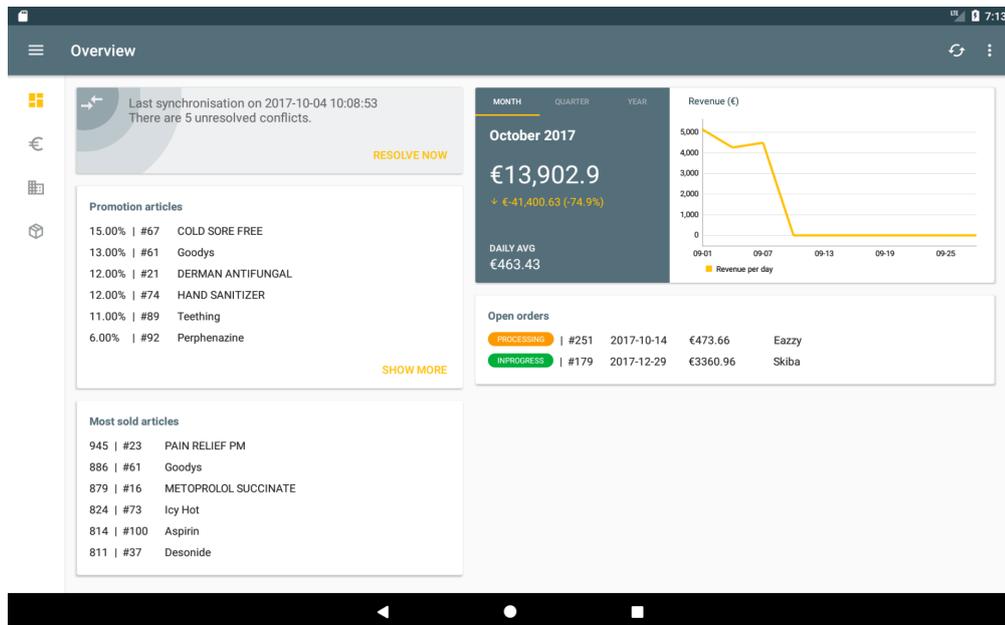


Figure 6.1: Overview screen

M	decide to create an order
M	search for the order screen icon in the small navigation drawer
P	point at and tap on the order icon in the small navigation drawer
R	system loads the order screen

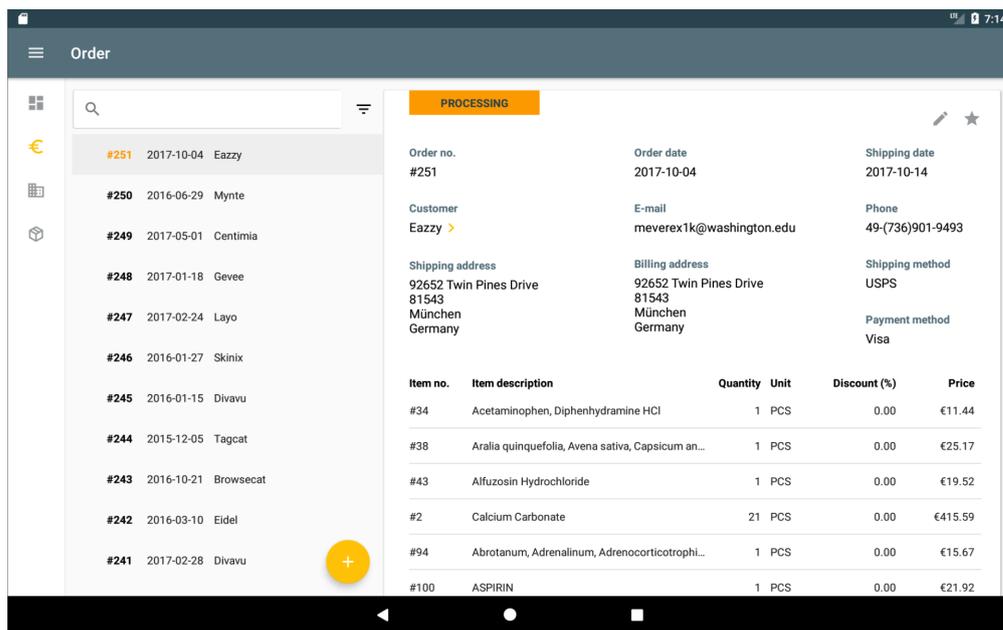


Figure 6.2: Order screen

M	search for the FAB
P	point at and tap on the FAB

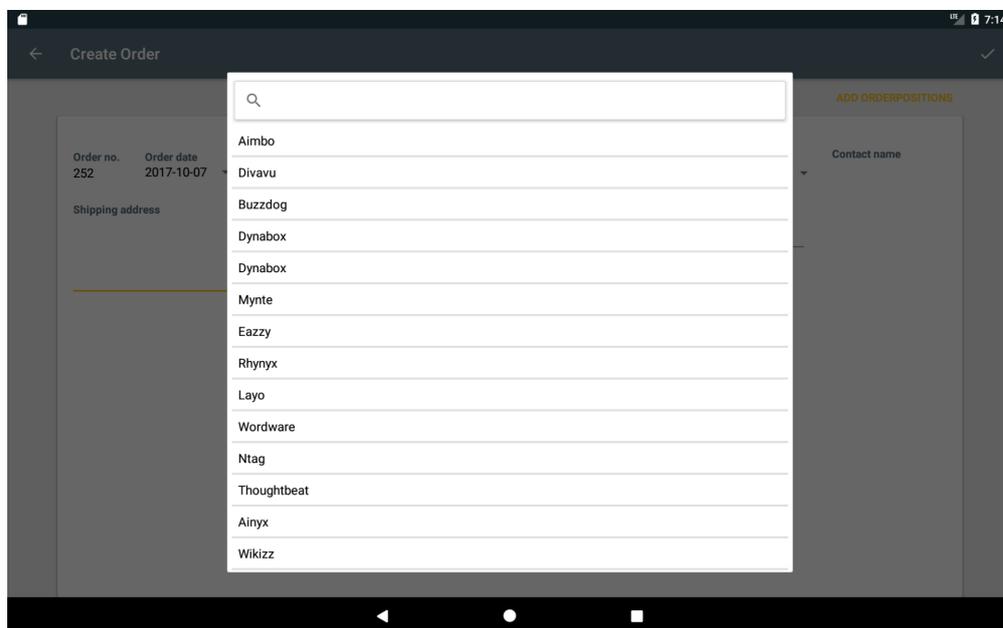


Figure 6.3: Screen for choosing the customer

6. EVALUATION

M	remember to select the customer
M	search for the search-box
P	point at and tap on the search-box
R	system opens the keyboard
M	remember the customer name
4T	type the first four letters of the customer name
M	search for the customer in the filtered list
P	point at and tap on the customer name
R	system loads the customer location popup

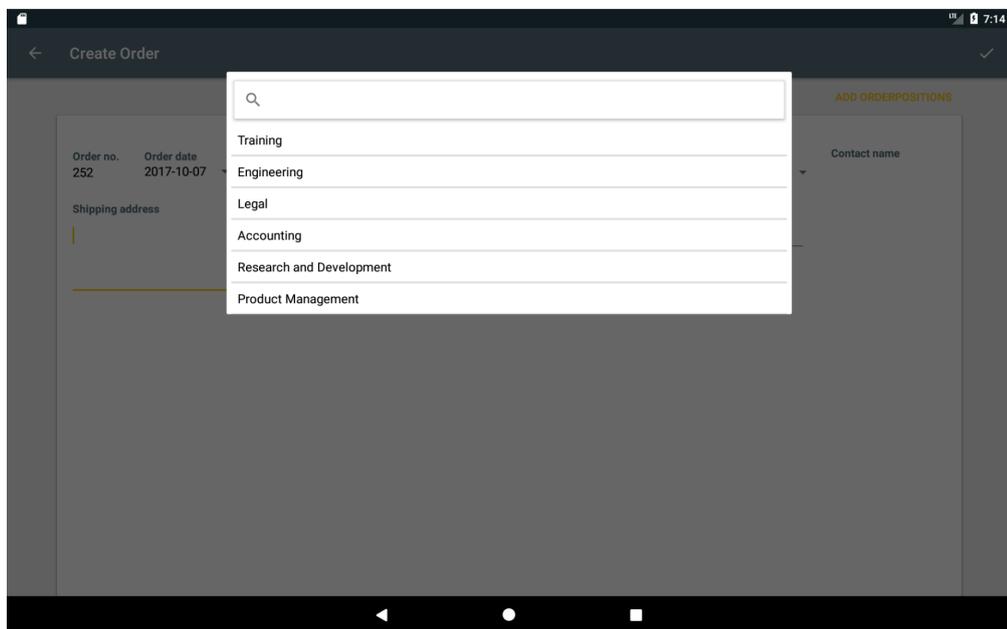


Figure 6.4: Screen for choosing the customer location

M	remember the customer location
M	search for the customer location
P	point at and tap on the customer location
R	system closes the popup and automatically loads all information in the correct fields

The screenshot shows a mobile application interface for creating an order. At the top, there is a header bar with a back arrow on the left, the text 'Create Order' in the center, and a checkmark on the right. Below the header, there is a form with several fields. The fields are organized into two columns. The left column contains 'Order no.' (252), 'Order date' (2017-10-07 with a dropdown arrow), 'Shipping date' (2017-10-17 with a dropdown arrow), 'Shipping address', and 'Billing address'. The right column contains 'Customer' (Dynabox with a dropdown arrow), 'Customer location' (Legal with a dropdown arrow), 'Contact name' (Onofredo Tomicki), 'Shipping method' (USPS), and 'Payment method' (Credit). The shipping address and billing address are both '51 Arkansas Drive, 80804, München, Germany'. At the bottom right of the form area, there is a button labeled 'ADD ORDERPOSITIONS'. The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps icons.

Figure 6.5: General order information screen

M	searching for the shipping date field
P	point at and tap on the shipping date or the triangle on its' right side
R	system loads the datepicker
M	remember the shipping date
M	search for the correct date
P	point at and tap on the correct date
R	system closes the datepicker and refreshes the shipping date
M	checking the customer name
M	checking the customer location
M	checking the shipping location
M	checking the billing location
M	remember what shipping method to use
P	point at and tap on the shipping method field
R	system opens the keyboard
5T	tap 5 letters (average length of english words)
M	remember what payment method to use
P	point at and tap on the payment method field

6. EVALUATION

5T	tap 5 letters (average length of english words)
M	confirm that the first screen is finished
M	search for the "ADD ORDER POSITIONS" button
P	point at and tap on the "ADD ORDER POSITIONS" button
R	system changes view

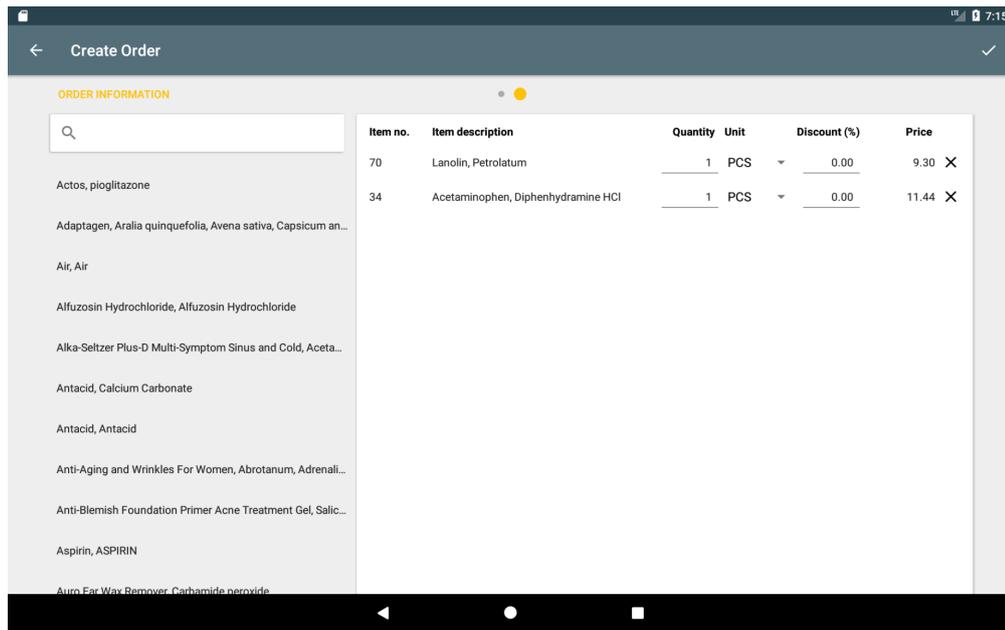


Figure 6.6: Screen for adding order positions

M	think about the article to add to the order
M	locate the search-box
P	point at and tap on the search box
4T	tap the first four letters of the article name
M	search for the article in the filtered list
P	point at and tap on the article
R	system adds the article as an order position
M	locate the order position
M	confirm the description of the ordered article
M	search for the quantity field

M	remember the quantity
P	point at and tap on the quantity field
T	type in the quantity
M	search for the unit field
M	remember the unit
P	point at and tap on the unit field
P	point at and tap on the unit
M	search for the discount field
M	remember the discount
P	point at and tap on the discount field
5T	type in the discount
M	check the price
M	confirm that nothing else needs to be added
M	search for the finish button
P	point at and tap on the finish button
R	system creates the order entry and switches to the order view

6.1.2 Keystroke-Level Model

In this section, the Keystroke-Level Model is applied to the order creation process of the other ERP system. The operators used are K (for keystroke) with the duration of 0.28 seconds, P (for pointing) with the duration of 1.1 seconds, M (for mental operator) with the duration of 1.35 seconds, H (for homing operator) with the duration of 0.4 seconds, R (for response time of the system) with the duration of 0.35 seconds. The values for the operators K, P, M and H are established by the original creators of KLM [CMN86]. The system operator R is variable, dependent on the response time of the ERP system.

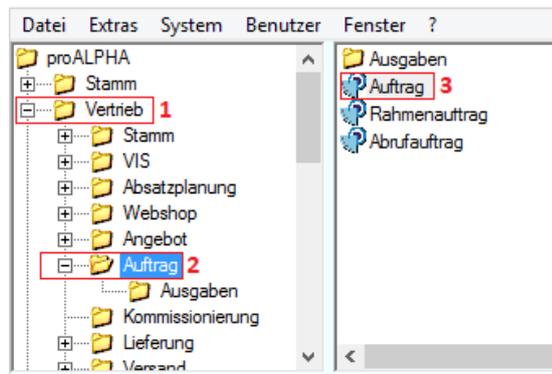


Figure 6.7: Folder structure of the ERP system

M	decide to create an order
M	search for sales folder (marked as 1 in Figure 6.7)
P	point at sales folder
K	click on sales folder
M	search for order folder (marked as 2 in Figure 6.7)
P	point at the order folder
K	click on the order folder
M	search for the order (marked as 3 in Figure 6.7)
P	point at the order
K	click on the order
R	system opens the order creation mask

The screenshot shows a software window titled "20 Test AT Auftrag". Inside, there is a sub-window titled "20 Test AT Neuanlage". The form contains the following fields and values:

- Belegnummer: 0
- Kunde: 22088888
- Name: Bachelorprototype Inc. (marked with '2')
- Suchbegriff: (empty)
- Selektion: (empty)
- Straße: (empty)
- Telefon: (empty)
- Ort: A 1110 Vienna
- Telefax: (empty)
- Belegdatum: 14.10.2017 Sa
- Sachbearbeiter: INSCHL
- Auftragsart: 99 Standard
- Preisgültigkeit: 07.10.2017 Sa
- Verteilergruppe: (empty)
- Wunschtermin: 07.10.2017 Sa
- Lagerort: ?
- Liefertermin: 07.10.2017 Sa (marked with '3')
- Dispobereich: 0 0
- KommisionsNr: 0
- Lieferbedingung: 0 keine
- Lieferrestriktion: keine Restriktionen
- Projekt: (empty)
- Versandart: 0 Selbstabholung
- Sprache: Deutsch
- Preisliste: (empty)
- steuerfrei:
- steuerpfl Ware:
- Kundenrabbatgruppe: (empty)
- Konfiguration:
- Rabatt sichtbar:

Figure 6.8: General order information mask

M	search for the "new order" button (marked as 1 in Figure 6.8)
P	point at the "new order" button
K	click on the "new order" button
R	system opens the general information mask
M	search for the customer field (marked as 2 in Figure 6.8)
P	point at the customer field
K	click on the customer field
2K	press strg + a to open the customer list
R	system opens the customer list
M	remember how to search
P	point at any customer
K	right click on that customer
P	point at "Suchen auswählen" in the context menu
K	click on "Suchen auswählen" in the context menu
R	system opens the search-box
M	search for the search-field

6. EVALUATION

P	point at the search-field
K	click on the search-field
H	home from the mouse to the keyboard
M	remember the customer name
4K	type the first 4 letters of the customer name
H	home from the keyboard to the mouse
M	search for the "OK" button
P	point at the "OK" button
K	click on the "OK" button
M	search for the customer in the filtered list
P	point at the customer in the filtered list
2K	double-click on the customer in the filtered list
R	system loads the customer in the general order information
M	search for the shipping date field (marked as 3 in Figure 6.8)
P	point at the shipping date field
K	click on the shipping date field
H	home from the mouse to the keyboard
M	remember the shipping date
8K	type the shipping date
H	home from the keyboard to the mouse
M	remember to click on "Neuanlage"
P	point at the "Neuanlage" button
K	click on the "Neuanlage" button
R	system goes back to the general order mask

Figure 6.9: Order creation mask

M	remember to add the contact name
M	search for the contact name field (marked as 1 in Figure 6.9)
P	point at the contact name field
K	click on the contact name field
H	home from the mouse to the keyboard
10K	type the contact name in the field
H	home from the keyboard to the mouse
M	decide to add an article
M	search for “Funktion” in the navigation bar (marked as 2 in Figure 6.9)
P	point at “Funktion” in the navigation bar
K	click on “Funktion” in the navigation bar
M	search for “Positionen” in the dropdown menu
P	point at “Positionen” in the dropdown menu
K	click on “Positionen” in the dropdown menu
R	system opens the order position mask

6. EVALUATION

The screenshot shows the SAP Order position screen for '20 Test AT Auftragspositionen 32000246'. The interface includes a menu bar (Datei, Funktion, Ansicht, Extras, Stamm, Info, ?) and a toolbar. The main area contains several input fields and labels:

- Position: 1, 0
- Teil: 92300071 (highlighted with a red box and '1')
- Kreditlimit: 0,00 EUR
- OP Saldo: 0,00 EUR
- Endbetrag: 0,00 EUR
- Bezeichnung: Material- u. IT-Beratung diverser Teil
- Lagerort: ?
- Menge: 0 (highlighted with a red box and '2')
- Stk (highlighted with a red box and '3')
- stat Warennummer: (empty)
- Präferenz:
- Wareneinsatz:
- reservierte Menge: 0 Stk
- gelieferte Menge: 0 Stk
- stornierte Menge: 0 Stk
- Einzelpreis: 0,00 E Stk U
- Fracht: 0,00 U
- Zuschlag: 0,00 U
- Kommissionsnummer: 32000246
- Verkaufspreis: 0,00
- Lieferbedingung: 0 keine
- % R/Z: 0,00 % (highlighted with a red box and '4')
- Gewicht: 0,000000 / Stk
- Verpackungsgewicht: 0,0000 / Stk
- Wunschtermin: 07.10.2017 Sa
- Liefertemin: 07.10.2017 Sa
- Gesamtpreis: 0,00 EUR
- Transportzeit: 0 KTg 06.10.2017 Fr
- NettoExpstoffgew: 00,00000
- autom Steuerfindung: 20,00%

Figure 6.10: Order position screen

M	search for the “Teil” field (marked as 1 in Figure 6.10)
P	point at the “Teil” field
K	click on the “Teil” field
2K	press strg + a to open the article list
R	system opens the article list
M	remember how to search
P	point at any article
K	right-click on that article
M	search for the “suchen auswählen” in the context menu
P	point at the “suchen auswählen” in the context menu
K	click on the “suchen auswählen” in the context menu
R	system opens the search-box

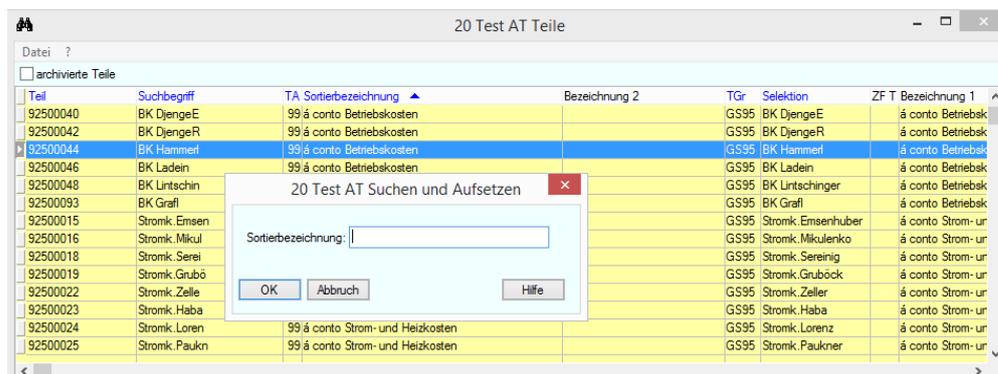


Figure 6.11: Article search screen

H	home from the mouse to the keyboard
M	remember the articles' name
4K	type the first 4 letters of the article
H	home from the keyboard to the mouse
M	search for the "OK" button
P	point at the "OK" button
K	click on the "OK" button
M	search for the article in the filtered article list
P	point at the article in the filtered article list
2K	double-click on the article in the filtered article list
R	system adds the article as an order position and goes back to the order position mask
K	press enter to finalise the article
M	search for the quantity field (marked as 2 in Figure 6.10)
P	point at the quantity field
K	click on the quantity field
M	remember the quantity
K	type in the quantity (only single digit for this comparison)
M	search for the unit field (marked as 3 in Figure 6.10)
P	point at the unit field
K	click on the unit field
M	remember the unit

P	point at the unit
K	click on the unit
M	search for the discount field (marked as 4 in Figure 6.10)
P	point at the discount field
K	click on the discount field
M	remember the discount for the article
H	home from the mouse to the keyboard
5K	type in the discount
H	home from the keyboard to the mouse
M	search for the save-icon
P	point at the save-icon
K	click on the save-icon
M	confirm that the order is done
M	search for the close-button
P	point at the close-button
K	click on the close-button
M	search for the save-button
P	point at the save-button
K	click on the save-button

6.1.3 Result

This section compares the results of the FLM and the KLM from the previous sections. The first table, Table 6.1, shows the results of the FLM, that was applied on the application. The second table, Table 6.2, shows the results of the KLM, that was applied on the ERP system. In both systems was the order creation process executed. Furthermore, the same information was added in each process. Therefore, these two produce the same outcome, with differentiation in steps, the amount of steps and time needed for it. It is important to note, that the results show the execution time of the task, when the user does not make any mistakes.

The results show a major difference in the amount of steps that are needed and the total duration of the order creation process. The application, that was developed

Operator	Number of occurrences	Total duration (seconds)
Tapping	24	7.44
Pointing	17	7.48
Mental	34	45.9
Response	10	3.5
Sum	85	64.32

Table 6.1: The results of the FLM

Operator	Number of occurrences	Total duration (seconds)
Keystrokes	66	18.48
Pointing	27	29.7
Mental	35	47.25
Homing	10	4
Response	10	3.5
Sum	148	102.93

Table 6.2: The results of the KLM

over the course of this bachelor thesis, outperforms the ERP system in every aspect. This shows, that the application can be a huge improvement in the order creation process. Since this thesis focus is on the field sales workers, this mobile application not only immensely reduces the communication between the field workers and in-house workers, but also decreases the time that is needed to create the actual order entry itself.

6.2 Cognitive Walkthrough

The goal of this evaluation is to review the usability of the main process, the order creation. For this, Cognitive Walkthrough as described in Section 4.3.2 has been used. In the following, the critical path will be thoroughly investigated, focusing on four question, which are:

- Will the user try to achieve the right effect (to get closer to the goal)?
- Is the interface element for this action perceptible or visible?

- Is there a clear connection between interface element and action? Does the user understand that this action is triggered by this interface element?
- Does the user get (sufficient) feedback? Does the user recognise, after executing an action, that he did the right thing?

Each of these questions will be answered with either 'yes', 'okay', or 'no'. Where 'yes' means that there are no problems, 'okay' that the action can be performed but with uncertainties, and 'no' that there are serious troubles with performing this step.

If there any problems found, those will be described in detail too.

6.2.1 Step-by-Step analysis

In the most likely scenario, the user will start at the order screen to create an order. Therefore, this will be the starting point for the evaluation.

Step 1: Opening the creation screen

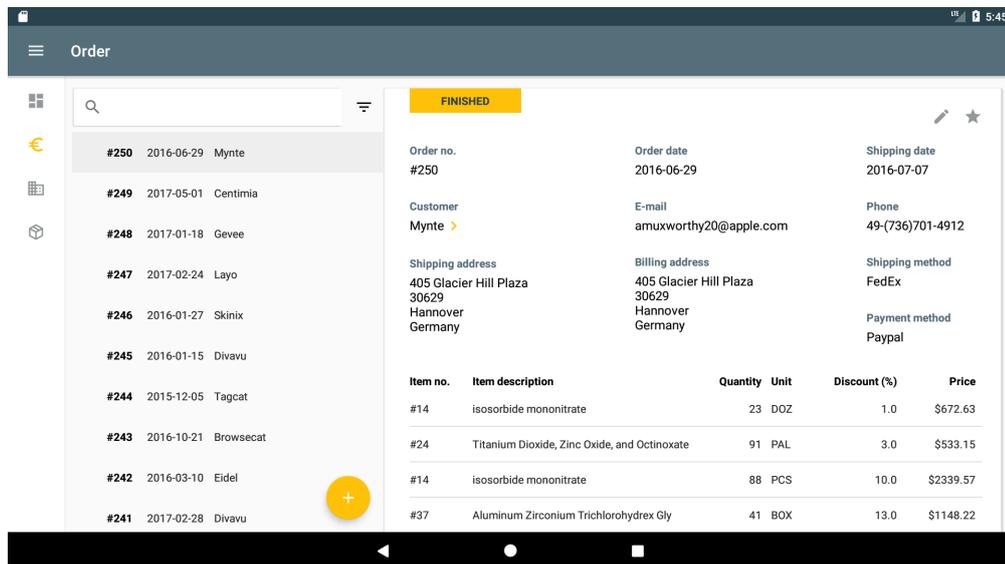


Figure 6.12: Step 1: Opening the creation screen

The first step is to open the order creation screen. Therefore, the user has to press the FAB.

- Effect? [yes]
- Visibility? [yes]
- Connection? [yes]
- Feedback? [yes]

Step 2: Picking the customer

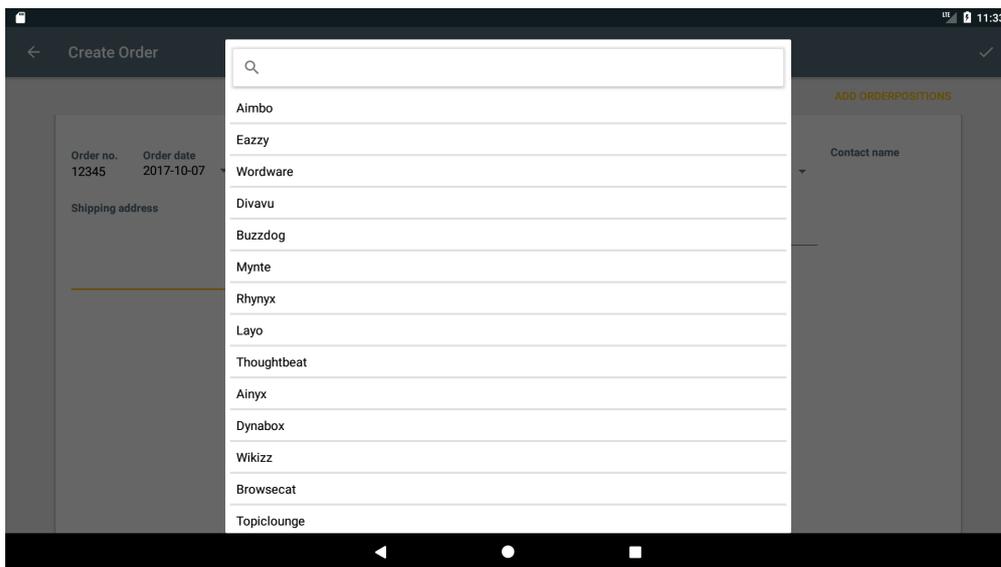


Figure 6.13: Step 2: Picking the customer

Right when the order creation screen has been opened, a list of customers pops up. Here the user has to pick the desired one.

- Effect? [yes]
- Visibility? [yes]
- Connection? [okay] While the user knows that these are customers, it is not perfectly clear that this screen is used to pick one. Therefore, a title "Pick customer" should be added to the list.
- Feedback? [yes]

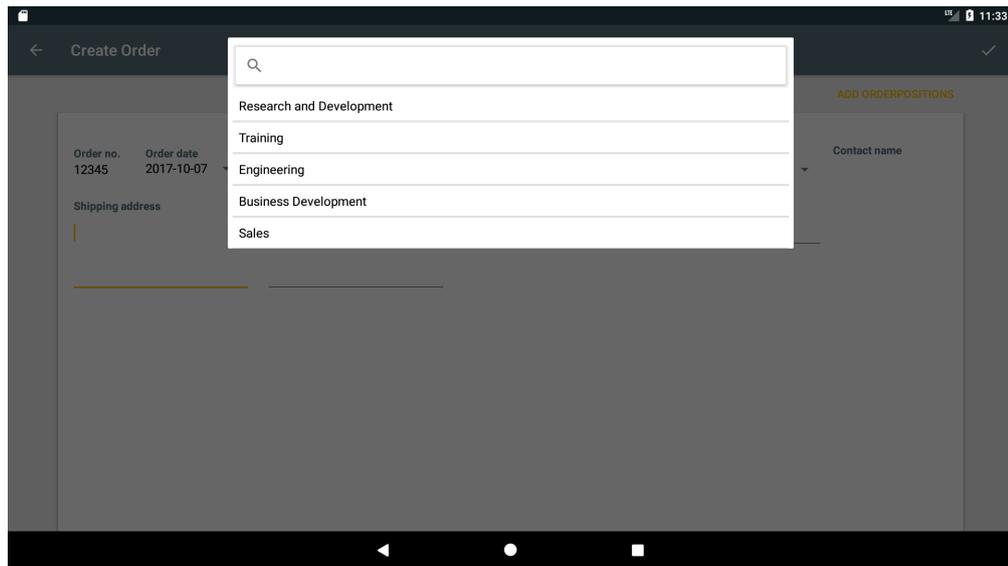


Figure 6.14: Step 3: Picking the customer location

Step 3: Picking the customer location

After the customer has been selected, a list of customer locations pops up. Here the user has to pick the desired one.

- Effect? [yes]
- Visibility? [yes]
- Connection? [okay] While the user knows that these are customer locations, it is not perfectly clear that this screen is used to pick one. Therefore, a title "Pick customer location" should be added to the list.
- Feedback? [yes]

Step 4: Checking the header information

After the customer and customer location have been picked, the information of these is automatically loaded into the screen. The user now has the possibility to check them and perform any needed changes. Typically, no changes have to be performed, except for changing the shipping date.

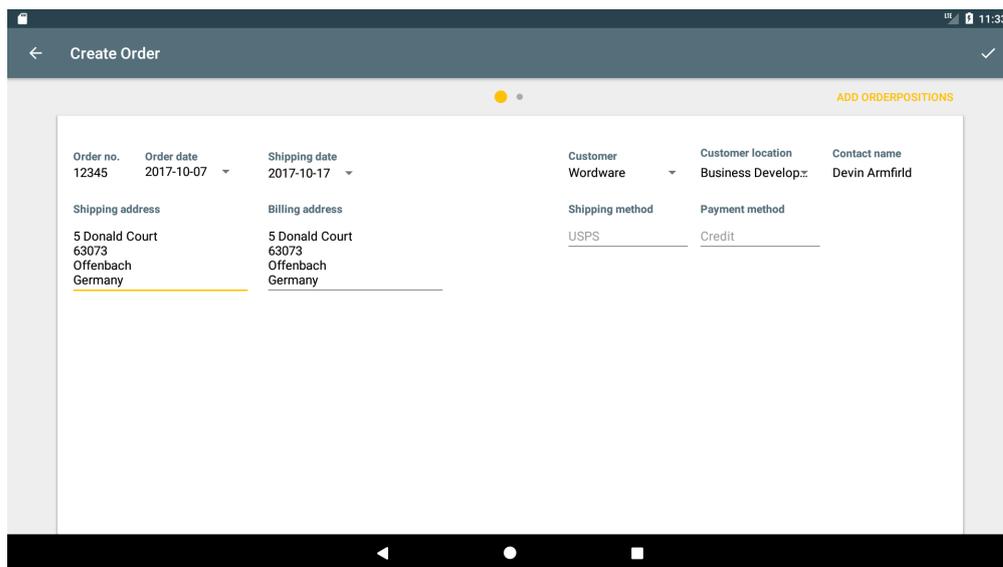


Figure 6.15: Step 4: Checking the header information

- Effect? [yes]
- Visibility? [yes]
- Connection? [yes]
- Feedback? [yes]

Step 5: Changing the shipping date

To change the shipping date the user has to click the date. This possibility is indicated by the arrow on the end. Once clicked, a standard date picker is opened, where the date can be set and confirmed.

- Effect? [yes]
- Visibility? [yes]
- Connection? [yes]
- Feedback? [yes]

Step 6: Navigation to the order positions

Since all header information is added, the user now navigates to the screen for adding order positions. This is achieved typically by clicking the 'ADD ORDER-

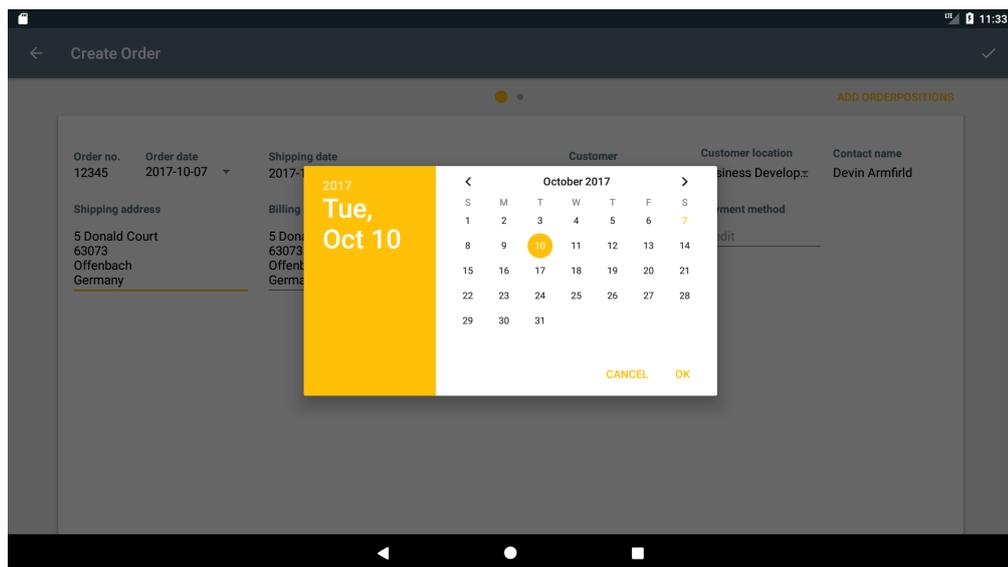


Figure 6.16: Step 5: Changing the shipping date

POSITIONS' button. Another way would be to swipe to the left, to get to the next screen, which is indicated by the indicator dots in the top-center area.

- Effect? [yes]
- Visibility? [yes]
- Connection? [yes]
- Feedback? [yes]

Step 7: Adding order positions

Once in the order position screen, the user is able to pick articles from the left side and by clicking, adding them to the order.

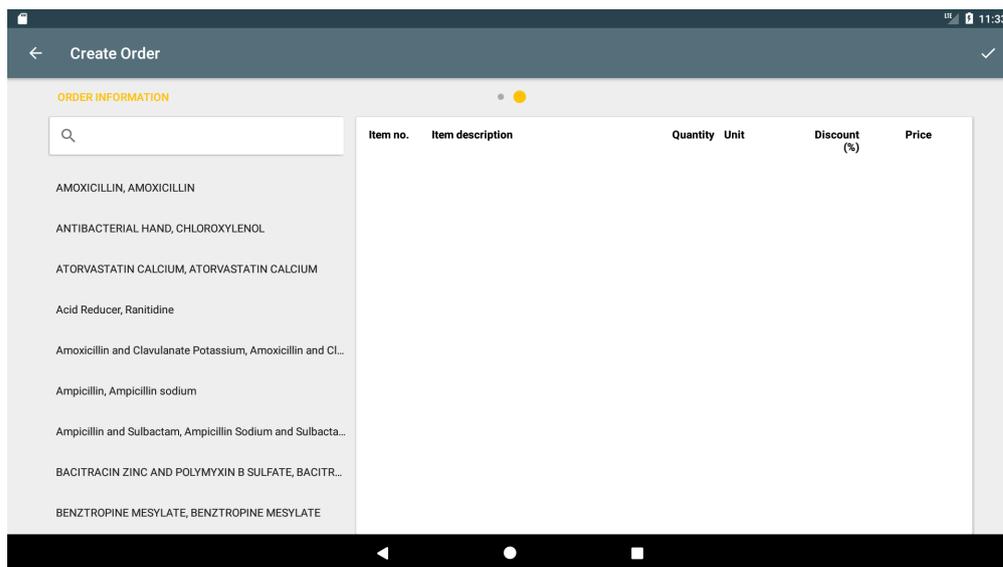


Figure 6.17: Step 7: Adding order positions

- Effect? [okay] While the user knows that he is in the screen for adding articles to the order, it could be not perfectly clear that he therefore has to click an item from the left side. Therefore, it could mean that he first just tries clicking it, but not knowing for sure that this means adding it to the order.
- Visibility? [yes]
- Connection? [okay] As described for the effect, the connection between clicking an item and adding it to the order could not be clear at first glance.
- Feedback? [yes]

Step 8: Adapting the order positions

After an item has been added to the order, it appears on the right side. Here the user can adapt the amount, the unity and the discount. The fields that are changeable are either presented as standard Edit-text fields, or indicate this with the arrow at the end.

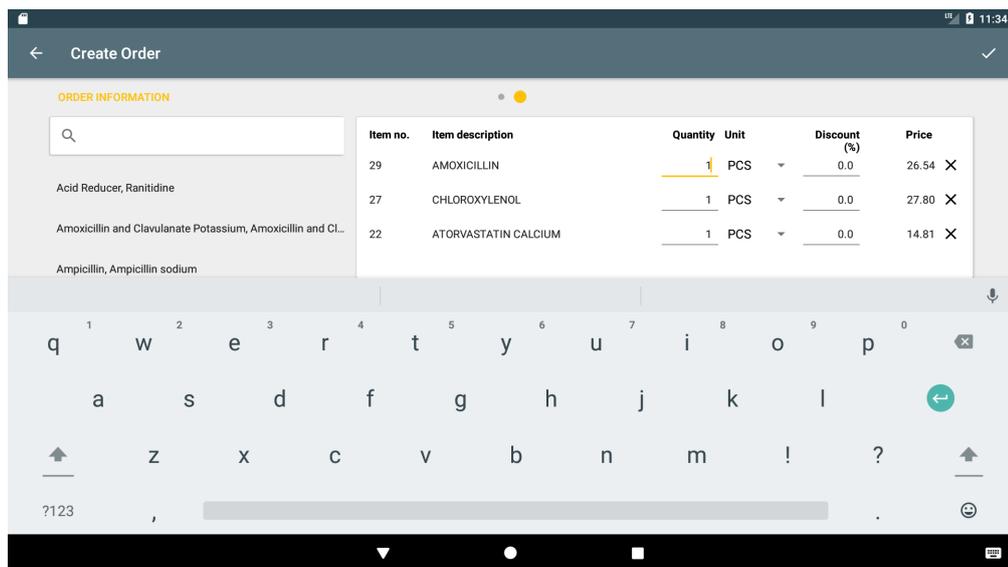


Figure 6.18: Step 8: Adapting the order positions

- Effect? [yes]
- Visibility? [yes]
- Connection? [yes]
- Feedback? [yes]

Step 9: Confirming the order

Finally, the user confirms the creation of the order, by clicking the check mark at the top right. Once clicked, this screen closes and the just created order is opened.

- Effect? [yes]
- Visibility? [yes]
- Connection? [yes]
- Feedback? [yes]

6.2.2 Results

The evaluation did not reveal any major problems with the usability of the order creation process. This shows that all uncertainties already have been found during the conception and implementation phase, using the User-Centered Design approach.

Though, some possible improvements have been uncovered, like adding a textual description when picking either the customer or the customer location. Other than that, the Cognitive Walkthrough also showed that the standard behaviour of lists in Android is, during the first usage, not perfect. Since listitems do not indicate in anyway that they are clickable. Though, this is only a problem when the element is interacted with the first time. Since after that, the user already knows that these are clickable. Nevertheless, a possible improvement could be to indicate the 'clickability' by animating the background of an item when the screen loads, showing the same animation as when the item later is clicked. Another possible solution would be to add a textual description at the top, like "Click an item to add it to the order", to make sure the user knows how to add an article as order position.

Conclusion and Future Work

Nowadays, the importance of mobile devices in businesses is undeniable. For a company to stay competitive they have to adapt to this trend, which means also changing their business processes. This need and the possible improvements that mobile devices, with adapted business processes, possess, in the field sales area, have been researched in this thesis.

First, this work analysed the typical sales process in Section 2.1, with a special focus on the order creation process. With this as foundation and by taking the new interaction possibilities, that mobile devices offer as described in Section 3.1, into consideration, an improved version was developed (Section 5.1). This adapted version allowed for a reduction of the time needed to create an order more than one third. In Section 6.1, by using the evaluation method FLM to measure the efficiency for the application and comparing it to the results that the KLM provided for another ERP-System, an improvement of 38% could be shown. This means that the time a user needs to create an order in an ERP-System could be reduced to two thirds of the original time.

Furthermore, unnecessary communication steps between the in-house and field workers could be removed, because the field sales team now has more information directly accessible to them. In detail, this is described in Section 5.1, where the new, improved version is shown in Figure 5.1, which, when compared to the original sales process (Figure 2.2, makes the reduced communication apparent. This not

only increases the efficiency of the field sales team but also allows the in-house workers to focus more on their job, reducing their overall workload. Moreover, the redundant intermediate step of noting down the orders on paper first is not needed anymore.

This work also showed that, by following the User-Centered Design approach, it is possible to create such an application, that even novice users are able to adapt easily to it and use it with satisfaction. This is especially apparent when looking at the evaluation steps, presented in Section 5.3.5, which show a steady improvement for each new version. Moreover, using this design idea allowed for an efficient implementation process, by letting user feedback flow in at each step. This resulted in fast development cycles and meant that no big issues had to be solved at the end. Which also was shown in the evaluation with the Cognitive Walkthrough (Section 6.2), that confirmed that no usability issues remained.

Though, the many evaluations with the target users also revealed possible points of improvement, which were not in the scope of this thesis. Many pointed out that the ability to also administer their customers information on the go would be really helpful. This shows that not only the sales process itself and with this the ERP-System, benefits from the usage of mobile applications, but also that CRM-Systems could be used in such a way. Therefore, the next step could be to further explore the possibilities a CRM-System offers and to check which parts could be beneficial in mobile usage.

Since this prototype is fully functional and the company, where the evaluation was conducted, already showed interest in using this application, further steps will be taken to add more functions, like parts of a CRM-System. Furthermore, a proper authorization schema and mechanism has to be established, to allow this application to be used in live operation. Another important topic will be securing the communication over the internet, by encrypting the exchanged data. Since this was already considered when developing the synchronisation process, this can be easily achieved, by extending the given TCP channel.

List of Figures

2.1	Simplified business process, based on [ACS96]	6
2.2	General order creation process, based on [JB04][Gud99][Hac98]	12
2.3	A general model of a system, based on [CC01]	14
2.4	ERP base modules, based on [Gra11]	15
2.5	Integration in a hierarchic structure, based on [Mei11]	17
2.6	Model representing a case handled by the three strategies, based on [OV96][OV11][Fra06][LKC04]	25
3.1	Material Design elements [Inca]	35
4.1	User-centered design process, based on [Jok02]	40
5.1	Adapted order creation process, based on Figure 2.2	47
5.2	Synchronisation architecture	50
5.3	Step 1: Getting master data from the server	51
5.4	Step 2: Checking for possible conflicts	52
5.5	Step 3: Sending local changes to the server	53
5.6	Sketch of the overview screen	64
5.7	Sketch of the article screen	64
5.8	The results for the general and navigation related questions	66
5.9	The use case rating	66
5.10	Overview screen of the wireframe	67
5.11	Order screen of the wireframe	67
5.12	Order creation process of the wireframe (part 1)	68
5.13	Order creation process of the wireframe (part 2)	68

5.14	The results for the general and navigation related questions	70
5.15	The use case rating	70
5.16	Overview screen of the mockup	71
5.17	Order screen of the mockup	71
5.18	Order creation process of the mockup (part 1)	72
5.19	Order creation process of the mockup (part 2)	72
5.20	The results for the general and navigation related questions	74
5.21	The use case rating	74
5.22	Overview screen of the prototype	75
5.23	Order screen of the prototype	75
5.24	Order creation process of the prototype (part 1)	76
5.25	Order creation process of the prototype (part 2)	76
6.1	Overview screen	78
6.2	Order screen	79
6.3	Screen for choosing the customer	79
6.4	Screen for choosing the customer location	80
6.5	General order information screen	81
6.6	Screen for adding order positions	82
6.7	Folder structure of the ERP system	84
6.8	General order information mask	85
6.9	Order creation mask	87
6.10	Order position screen	88
6.11	Article search screen	89
6.12	Step 1: Opening the creation screen	92
6.13	Step 2: Picking the customer	93
6.14	Step 3: Picking the customer location	94
6.15	Step 4: Checking the header information	95
6.16	Step 5: Changing the shipping date	96
6.17	Step 7: Adding order positions	97
6.18	Step 8: Adapting the order positions	98

List of Tables

5.1	Available commands for the synchronisation	48
6.1	The results of the FLM	91
6.2	The results of the KLM	91

Bibliography

- [134] ISO 13407. Iso 13407. <https://www.iso.org/obp/ui/#iso:std:iso:13407:ed-1:v1:en>. Accessed: 2017-09-30.
- [185] ISO 18529. Iso 18529. <https://www.iso.org/obp/ui/#iso:std:iso:tr:18529:ed-1:v1:en>. Accessed: 2017-09-30.
- [Abd11] Evgeniy Abdulin. Using the keystroke-level model for designing user interface on middle-sized touch screens. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '11, pages 673–686, New York, NY, USA, 2011. ACM.
- [ABES14] Sascha Alpers, Christoph Becker, Esmahan Eryilmaz, and Thomas Schuster. *A Systematic Approach for Evaluation and Selection of ERP Systems*, pages 36–48. Springer International Publishing, Cham, 2014.
- [Ack71] Russell L. Ackoff. Towards a system of systems concepts. *Management Science (pre-1986)*, 17(11):661, 07 1971. Copyright - Copyright Institute for Operations Research and the Management Sciences Jul 1971; Zuletzt aktualisiert - 2010-06-07; CODEN - MNSCDI.
- [ACS96] Kim Fischer Axel C. Schwickert. *Der Geschäftsprozess als formaler Prozess - Definition, Eigenschaften und Arten*. Gabler Verlag, 1996.
- [ASS02] Saligrama Agnihotri, Nagaraj Sivasubramaniam, and Donald Simmons. Leveraging technology to improve field service. *International Journal of Service Industry Management*, 13(1):47–68, mar 2002.

- [BCP01] Artur Baldauf, David W. Cravens, and Nigel F. Piercy. Examining the consequences of sales management control strategies in european field sales organizations. *International Marketing Review*, 18(5):474–508, oct 2001.
- [BOB05] Jared Benson, Ken Olewiler, and Nancy Broden. Typography for mobile phone devices: The design of the qualcomm sans font family. In *Proceedings of the 2005 Conference on Designing for User eXperience, DUX '05*, New York, NY, USA, 2005. AIGA: American Institute of Graphic Arts.
- [BP08] Timo Botzenhardt and Key Pousttchi. Analyzing the benefits of mobile enterprise solutions using the example of dispatching processes. In *2008 7th International Conference on Mobile Business*. IEEE, jul 2008.
- [BT93] Izak Benbasat and Peter Todd. An experimental investigation of interface design alternatives: icon vs. text and direct manipulation vs. menus. *International Journal of Man-Machine Studies*, 38(3):369 – 402, 1993.
- [BvG02] Markus Beier and Vittoria von Gizycki, editors. *Usability*. Springer Berlin Heidelberg, 2002.
- [CC01] Graham Curtis and David Cobham. *Business Information Systems: Analysis, Design, and Practice*. Financial Times Management, 2001.
- [CFH97] Michael D Corry, Theodore W Frick, and Lisa Hansen. User-centered design and usability testing of a web site: An illustrative case study. *Educational Technology Research and Development*, 45(4):65–76, 1997.
- [Cli15] Ian G. Clifton. *Android User Interface Design: Implementing Material Design for Developers (Usability)*. Addison-Wesley Professional, 2015.
- [CLS08] Yen-ning Chang, Youn-kyung Lim, and Erik Stolterman. Personas: From theory to practices. In *Proceedings of the 5th Nordic Conference*

on Human-computer Interaction: Building Bridges, NordiCHI '08, pages 439–442, New York, NY, USA, 2008. ACM.

- [CMN86] Stuart K. Card, Thomas P. Moran, and Allen Newell. *The Psychology of Human-Computer Interaction*. Lawrence Erlbaum Associates, 1986.
- [Com13] Douglas E. Comer. *Internetworking with TCP/IP Volume One: 1*. Pearson, 2013.
- [CRCN14] Alan Cooper, Robert Reimann, David Cronin, and Christopher Noessel. *About face: the essentials of interaction design*. John Wiley & Sons, 2014.
- [CTC05] David C. Chou, Hima Bindu Tripuramallu, and Amy Y. Chou. Bi and erp integration. *Information Management & Computer Security*, 13(5):340–349, 2005.
- [Dav98] Thomas H. Davenport. Putting the enterprise into the enterprise system. *Harvard Bus. Rev.*, 76(4):121–131, July 1998.
- [DJ03] A. Dabkowski and A.M. Jankowska. Comprehensive framework for mobile ERP system. In *14th International Workshop on Database and Expert Systems Applications, 2003. Proceedings*. IEEE Comput. Soc, 2003.
- [dSC08] Marco de Sá and Luís Carrico. Lessons from early stages design of mobile applications. In *Proceedings of the 10th international conference on Human computer interaction with mobile devices and services*, pages 127–136. ACM, 2008.
- [Fra06] Lars Frank. *Architecture for Distributed ERP Systems*, pages 71–83. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
- [Gad13] Andreas Gadatsch. *Grundkurs Geschäftsprozess-Management*. Springer Nature, 2013.

- [Gal07] W.O. Galitz. *The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques*. Wiley Desktop Editions. Wiley, 2007.
- [GHOS96] Jim Gray, Pat Helland, Patrick O’Neil, and Dennis Shasha. The dangers of replication and a solution. *SIGMOD Rec.*, 25(2):173–182, June 1996.
- [Gra11] Peter A. Grammer. *Der ERP-Kompass - ERP-Projekte zum Erfolg führen*. MITP, Bonn, 2011. aufl. edition, 2011.
- [Gro10] Norbert Gronau. *Enterprise Resource Planning: Architektur, Funktionen und Management von Erpsystemen: Architektur, Funktionen und Management von ERP-Systemen (German Edition)*. Oldenbourg Wissenschaftsverlag, 2010.
- [Gud99] Timm Gudehus. *Logistik*. Springer Berlin Heidelberg, 1999.
- [Hac98] Oliver Hackl. *Mitarbeiter im Verkaufsaußendienst*. Deutscher Universitätsverlag, 1998.
- [Heg03] Marcus Hegner. *Methoden zur Evaluation von Software*, volume 29. Informationszentrum Sozialwissenschaften, Bonn, 2003.
- [Hen15] Angelina Henschel. *Konzept und Design einer grafischen Benutzeroberfläche eines datenzentrierten Prozess-Management-Systems*. PhD thesis, Ulm University, 2015.
- [HIP11] HAJO HIPPNER. *Grundlagen des CRM. Strategie, Geschäftsprozesse und IT-Unterstützung*. Gabler Verlag, 2011.
- [Hof15] Gunter Hofbauer. *Professionelles Vertriebsmanagement Der Prozessorientierte Ansatz Aus Anbieter- Und Beschaffersicht (German Edition)*. Publicis MCD Werbeagentur GmbH, 2015.
- [HSS07] Kevin B. Hendricks, Vinod R. Singhal, and Jeff K. Stratman. The impact of enterprise systems on corporate performance: A study of

- ERP, SCM, and CRM system implementations. *Journal of Operations Management*, 25(1):65–82, jan 2007.
- [HV15] Hartmut Hoehle and Viswanath Venkatesh. Mobile application usability: Conceptualization and instrument development. *Mis Quarterly*, 39(2), 2015.
- [Inca] Google Inc. Material design guidelines. <https://material.io/guidelines/>. Accessed: 2017-09-30.
- [Incb] Google Inc. Material design principles. <https://material.io/guidelines/#introduction-principles>. Accessed: 2017-09-30.
- [JB04] Reinhard Schütte Jörg Becker. *Handelsinformationssysteme*. Moderne Industrie, 2004.
- [JG07] Christian Jetter and Jens Gerken. A simplified model of user experience for practical application. In *2nd COST294-MAUSE*, pages 106–111, 2007.
- [JK96] Bonnie E. John and David E. Kieras. Using goms for user interface design and evaluation: Which technique? *ACM Trans. Comput.-Hum. Interact.*, 3(4):287–319, December 1996.
- [Jok02] Timo Jokela. Making user-centred design common sense: striving for an unambiguous and communicative ucd process model. In *Proceedings of the second Nordic conference on Human-computer interaction*, pages 19–26. ACM, 2002.
- [JSvdBG04] Monique W.M. Jaspers, Thiemo Steen, Cor van den Bos, and Maud Geenen. The think aloud method: a guide to user interface design. *International Journal of Medical Informatics*, 73(11):781 – 795, 2004.
- [KG06] Irina Kondratova and Ilia Goldfarb. *Cultural Interface Design: Global Colors Study*, pages 926–934. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

- [Kie99] David E Kieras. A guide to goms model usability evaluation using goms1 and glean3. *University of Michigan*, 313, 1999.
- [Kue78] Harald Kueppers. *Das Grundgesetz der Farbenlehre (DuMont Kunst-Taschenbuecher ; 65) (German Edition)*. DuMont, 1978.
- [Liba] ACM Digital Library. Acm digital library. dl.acm.org/. Accessed: 2017-09-30.
- [Libb] IEEE Xplore Digital Library. Ieee xplore digital library. ieeexplore.ieee.org/. Accessed: 2017-09-30.
- [LKC04] YoungSeok Lee, YounSoo Kim, and Hoon Choi. *Conflict Resolution of Data Synchronization in Mobile Environment*, pages 196–205. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.
- [LSR⁺15] Ahreum Lee, Kiburm Song, Hokyoung Blake Ryu, Jieun Kim, and Gyuhyun Kwon. Fingerstroke time estimates for touchscreen-based mobile gaming interaction. *Human Movement Science*, 44(Supplement C):211 – 224, 2015.
- [MB02] Martin Maguire and Nigel Bevan. *User Requirements Analysis*, pages 133–148. Springer US, Boston, MA, 2002.
- [Mei11] Vera Meister. *Grundlagen betrieblicher Anwendungssysteme*. Expert-Verlag GmbH, 2011.
- [Mos12] Christian Moser. User experience design. In *User Experience Design*, pages 1–22. Springer Berlin Heidelberg, jul 2012.
- [MSK10] Thomas Mahatody, Mouldi Sagar, and Christophe Kolski. State of the art on the cognitive walkthrough method, its variants and evolutions. *International Journal of Human–Computer Interaction*, 26(8):741–785, 2010.
- [NSS05] Fiona Fui-Hoon Nah, Keng Siau, and Hong Sheng. The value of mobile applications: A utility company study. *Commun. ACM*, 48(2):85–90, February 2005.

- [O’L04] Daniel E O’Leary. On the relationship between REA and SAP. *International Journal of Accounting Information Systems*, 5(1):65–81, may 2004.
- [OV96] M. Tamer Özsu and Patrick Valduriez. Distributed and parallel database systems. *ACM Comput. Surv.*, 28(1):125–128, March 1996.
- [OV11] M. Tamer Özsu and Patrick Valduriez. *Principles of Distributed Database Systems, Third Edition*. Springer New York, 2011.
- [Sch01] Anna Noakes Schulze. User-centered design for information professionals. *Journal of Education for Library and Information Science*, 42(2):116, 2001.
- [SH10] Martin Schrepp and Theo Held. Anwendung von GOMS-analysen und CogTool in der design-praxis. In *Mensch & Computer 2010*. OLDENBOURG WISSENSCHAFTSVERLAG, 2010.
- [SWW02] Sanjay K. Singh, Hugh J. Watson, and Richard T. Watson. Eis support for the strategic management process. *Decision Support Systems*, 33(1):71 – 85, 2002.
- [THK03] Frank Teuteberg, Jens Hilker, and Karl Kurbel. Anwendungsschwerpunkte im mobile enterprise resource planning: State-of-the-art und entwicklungspotenziale. *Mobile Commerce-Anwendungen und Perspektiven*", Köller Druck & Verlag, Bonn, pages 12–26, 2003.
- [Too] Material Design Color Tool. Material design color tool. <https://material.io/color/#!/?view.left=0&view.right=0>. Accessed: 2017-09-30.
- [vdHJS03] Maaïke van den Haak, Menno De Jong, and Peter Jan Schellens. Retrospective vs. concurrent think-aloud protocols: Testing the usability of an online library catalogue. *Behaviour & Information Technology*, 22(5):339–351, 2003.

- [Wat12] Richard T. Watson. *Information Systems*. CreateSpace Independent Publishing Platform, 2012.
- [Wil09] Ashley Williams. User-centered design, activity-centered design, and goal-directed design: A review of three methods for designing web applications. In *Proceedings of the 27th ACM International Conference on Design of Communication*, SIGDOC '09, pages 1–8, New York, NY, USA, 2009. ACM.
- [WPS⁺00] M. Wiesmann, F. Pedone, A. Schiper, B. Kemme, and G. Alonso. Understanding replication in databases and distributed systems. In *Proceedings of the The 20th International Conference on Distributed Computing Systems (ICDCS 2000)*, ICDCS '00, pages 464–, Washington, DC, USA, 2000. IEEE Computer Society.