

INVESTIGATING USABILITY ISSUES IN ERP SYSTEMS:  
THE CASES OF MULTI-UI TASKS AND UI VALIDATION

---

A Thesis

presented to

the Faculty of Natural and Applied Sciences

at Notre Dame University-Louaize

---

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Computer Science

---

by

HANINE EL HAYEK

DECEMBER 2018

©COPYRIGHT

By

Hanine El Hayek

2018

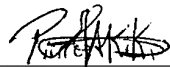
**All Rights Reserved**

Notre Dame University - Louaize Faculty  
of Natural and Applied Sciences  
Department of Computer Science

We hereby approve the thesis of

Hanine El Hayek

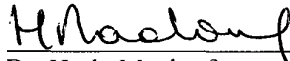
Candidate for the degree of Master of Science in Computer Science



---

Dr. Pierre A. Akiki

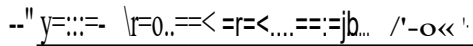
Supervisor, Chair



---

Dr. Hoda Maalouf

Committee Member



---

Dr. Marie Khair

Committee Member

## Thesis Release Form

I, Harime El Hayek, authorize Notre Dame University-Louaize to supply copies of my thesis to libraries or individuals on request.

**D** I \_\_\_\_\_, do not authorize Notre Dame University-Louaize to supply copies of my thesis to libraries or individuals on request.

 \_\_\_\_\_

Signature

5/2/2019  
Date

## **Acknowledgments**

I wish to thank my committee members Dr. Hoda Maalouf and Dr. Marie Khair for their precious time and effort.

I am greatly indebted to my thesis advisor, Dr. Pierre A. Akiki, for his contribution, effort, help, support, and guidance throughout the entire process of my thesis.

Sometimes saying thank you is not enough to express all the gratefulness I feel to my loving and supportive family. I am blessed to have that source of encouragement and inspiration in my life. I would like to thank them from the bottom of my heart for being always by my side, believing in me, giving me strength to chase my dreams and overcome all the obstacles that stood in my way.

And above all, praises to Almighty God for his guidance and blessings in my life.

# Table of Contents

Table of Contents .....	v
List of Figures.....	viii
List of Tables .....	x
List of Code Listings .....	xi
List of Abbreviations .....	xii
Abstract.....	xiii
Chapter 1: Introduction and Problem Definition .....	1
1.1 What are ERP Systems? .....	1
1.2 Introduction to the General Problem .....	2
1.3 Problem Definition .....	3
1.4 Research Objectives .....	3
1.5 Approach and Main Results .....	3
1.6 Thesis Organization .....	4
Chapter 2: Background and Motivation.....	5
2.1 Enterprise Departmentalization and Communication .....	5
2.2 ERP System Evolution and Revenues.....	?
2.3 ERP System Critical Success Factors .....	9
2.3.1 Appropriate Business and IT Legacy Systems .....	9
2.3.2 Business Plan and Vision .....	9
2.3.3 Business Process Reengineering .....	9
2.3.4 Change Management Culture and Program .....	10
2.3.5 Communication .....	10
2.3.6 ERP Teamwork and Composition .....	10
2.3.7 Monitoring and Evaluation of Performance .....	10
2.3.8 Project Champion .....	10
2.3.9 Project Management .....	11
2.3.10 Software Development, Testing and Troubleshooting.....	11
2.3.11 Top Management Support .....	11
2.4 ERP Risk Categories .....	11
2.4.1 People issues.....	12
2.4.2 Process risks .....	14
2.4.3 Technological risks.....	15

2.5 ERP Business Productivity Measures .....	17
2.5.1 Usability .....	17
2.5.2 Collaboration .....	18
2.5.3 Business Insight .....	18
2.5.4 Flexibility .....	18
2.5.5 Transactional Efficiency .....	18
2.5.6 Fmmiliarity .....	19
2.6 Previous Work on Usability Problems in ERP Systems.....	19
2.6.1 Navigation .....	20
2.6.2 Presentation .....	21
2.6.3 Task Support .....	22
2.6.4 Learnability.....	23
2.6.5 Customization .....	24

**Chapter 3: A Study that Investigates Usability Issues in the SAP ERP System.....25**

3.1 Study Design .....	25
3.1 Participants .....	26
3.2 Study Results-SAP Usability Problems.....	27
3.3 Threats to Validity .....	30
3.4 Existing Work on UI Composition.....	31
3.5 Existing Work on UI Validation .....	31
3.6 Research Motivation .....	33

**Chapter 4: A Study that Assesses the Usability Impact of Merging Multiple Uis Based on the Task.....34**

4.1 Study Design .....	34
4.2 Study Results .....	39

**Chapter 5: An Approach for Developing Model-driven Multi-target Validation.....44**

5.1 Problem with Error Messages .....	44
5.2 Study of Error Messages and their Proposed Validation Types.....	45
5.3 Proposed Validation Model Concepts .....	49
5.4 Multi-target Validation .....	51
5.4.1 Domain Model .....	51
5.4.2 Database .....	51
5.4.3 User Interface .....	52

5.4.4 Validation Model .....	55
5.5 Examples of Validation Rules .....	58
<b>Chapter 6: Conclusion .....</b>	<b>66</b>
6.1 Main Contributions .....	66
6.2 Possible Extensions and Future Work.....	67
<b>Bibliography .....</b>	<b>68</b>
<b>Appendix A: Questionnaires used in the Studies .....</b>	<b>72</b>
<b>Appendix B: Identified Error Validation Types .....</b>	<b>79</b>



# List of Figures

Figure 2-1 -Lack of communication between departments of an organization (Leon, 2013) .....	6
Figure 2-2- Information shared between departments in an enterprise (Leon, 2013) .....	6
Figure 2-3 - ERP evolution timeline .....	7
Figure 2-4- Estimation ofERP application revenues between 2006 and 2011 .....	8
Figure 2-5-Categories of Risks in ERP Systems.....	12
Figure 2-6 - Business Productivity Measurement Framework (Keystone Strategy, 2007) ..	17
Figure 2-7 - ERP Usability Criteria (Singh & Wesson, 2009) .....	19
Figure 3-1 -Thesis diagram .....	26
Figure 3-2- Demographic information of participants from accounting .....	27
Figure 3-3- Demographic information of participants from sales .....	27
Figure 4-1 -Organizing fields using Form Settings .....	35
Figure 4-2 -Adding fields with User-Defined Fields Management.....	36
Figure 4-3 -Initial SAP sales opportunity UI .....	37
Figure 4-4- Adapted sales opportunity UI .....	37
Figure 4-5 -Initial SAP invoice UI .....	38
Figure 4-6- Adapted invoice UI .....	38
Figure 4-7 -Time taken to complete the initial and adapted invoice UIs .....	40
Figure 4-8- Time taken to complete the initial and adapted opportunity UIs .....	41
Figure 4-9- SUS Scores for the initial and adapted invoice UIs .....	41
Figure 4-10- SUS Scores for the initial and adapted opportunity UIs .....	42
Figure 4-11 -Product reaction cards selected for initial and adapted invoice UIs .....	43
Figure 4-12 -Product reaction cards selected for initial and adapted opportunity UIs .....	43
Figure 5-1 -Total number oftimes the validation rules are used .....	46
Figure 5-2 -Total number of different validation rules used per UI .....	47
Figure 5-3- Number ofvalidation operations per rule.....	48
Figure 5-4-High-level concept class diagram .....	49
Figure 5-5- Validation rules class diagram .....	50
Figure 5-6- Domain model of the sales order .....	52
Figure 5-7 - Database diagram of the sales order .....	53

Figure 5-8- Example of sales order (extract) UI represented on multiple levels of abstraction: (a) task model, (b) AUI, (c) CUI, and (d) FUI .....	54
Figure 5-9- The comparison rule for the sales order extract example .....	55
Figure 5-10- Triggering the validation model on change and activation .....	56
Figure 5-11 -Mandatory rule example .....	58
Figure 5-12 - Restrict type and size rule example .....	59
Figure 5-13- Restrict value rule example .....	59
Figure 5-14- Regular Expression rule example .....	60
Figure 5-15- Range rule example .....	60
Figure 5-16-General models of query validation rules: (a) Exists rule (b) Sum rule .....	61
Figure 5-17- Exists rule example .....	62
Figure 5-18- Count rule example .....	63
Figure 5-19 - Count Distinct rule example .....	64
Figure 5-20- Sum rule example .....	65

## List of Tables

Table 2.1 -Usability problems related to navigation .....	21
Table 2.2 -Usability problems related to presentation .....	22
Table 2.3 -Usability problems related to task support .....	23
Table 2.4- Usability problems related to learnability .....	24
Table 2.5 -Usability problems related to customization .....	24
Table 3.1 -SAP Usability Problems Reported by Participants .....	28
Table 4.1 -Results of comparing adapted UIs to initialSAP UIs (1=adapted UI is much worse to 5=adapted UI is much better) .....	39
Table 5.1 -Unclear error messages .....	44
Table 5.2- Unhelpful error messages .....	45
Table 5.3 - Sample error messages with their validation types .....	46
Table 5.4- Descriptive statistics ofthe validation rules .....	48

## List of Code Listings

Listing 5.1 -Example of validation on a domain model class .....	56
Listing 5.2- Example of validation on a database.....	57
Listing 5.3- Example of validation on the UI .....	57

# List of Abbreviations

ERP: Enterprise Resource Planning

UI: User Interface

MRP: Material Requirement Planning

BOM: Bill of Materials

CRM: Customer Relationship Management

ISP: Internet Service Provider

UML: Unified Modelling Language

OCL: Object Constraint Language

SQL: Structured Query Language

PRC: Product Reaction Card

SUS: System Usability Scale

CTT: ConcurTaskTree

AUI: Abstract User Interface

CUI: Concrete User Interface

FUI: Final User Interface

# Abstract

Enterprise Resource Planning (ERP) systems are vital for managing any modern-day business. These systems aim to facilitate an organization's work by automating and integrating all business information and processes. However, ERP systems suffer from many problems that may cause an implementation failure. Many of these problems are related to usability, which is affected by the complexity of user interfaces (UIs). Usability can be defined in terms of an end-user's efficiency, effectiveness, and satisfactions.

A study was conducted in this thesis to inquire about the usability problems that ERP users face. The results showed that ERP usability is mostly affected by two issues. Although other issues were identified these two issues were the most common. The first issue is related to the cases where end-users are required to navigate through many UIs in order to accomplish a task. The second issue is related to the numerous incomprehensible error messages that stress out and sometimes mislead end-users.

The contribution of this thesis is divided into two parts, based on the two common usability issues that were identified by the previously mentioned study. The first part of the contribution involved conducting an empirical study to test the usability difference between accomplishing a task using multiple UI and a single adapted UI. The outcome showed that the single adapted UI saves time and users found it to be much better than the initial set of UIs. The second part of the contribution involved analyzing a sample of error messages from various SAP Business One UIs. Based on this analysis, ten types of validation rules were suggested. Then, the design of an approach for developing model-driven multi-target UI validation was presented. This approach is capable of executing validation on multiple levels including UIs, software classes, and databases.

Keywords: ERP System, User Interface, Usability, Validation

# Chapter 1: Introduction and Problem Definition

This chapter introduces the enterprise resource planning system along with the general problem and specific problems related to the usability of enterprise resource planning (ERP) systems. This chapter also defines the research objectives, and provides a summary of the approach, main results, and thesis organization.

## 1.1 What are ERP Systems?

An ERP system is an integrated software system that helps a company in managing its business resources efficiently and effectively. It aims to facilitate an organization's work by automating and integrating all business information and processes. ERP systems enable companies to share, access and analyze real-time data and practices from their different departments (Vaman, 2007). ERP systems manage major business operations such as: human resources, purchasing, inventory, production, sales, marketing, accounting, and finance. These main functional areas of an enterprise constitute the modules of ERP software. Some examples of popular ERP system providers include the following: SAP, Oracle, Microsoft, Sage, Infor, and Epicor (Marder, 2017).

The business benefits of an enterprise resource planning system were classified in a framework of five dimensions which are operational, managerial, strategic, IT infrastructure, and organizational (Shang & Seddon, 2000). These benefits consist of reducing cost and cycle time of activities in addition to improving productivity, quality, and customer service. ERP systems also enhance resource management, decision making, and performance control. These systems can also affect employee satisfaction.

## 1.2 Introduction to the General Problem

ERP systems are used by companies all over the world. Despite the advantages that ERP systems offer and their growing market, these systems suffer from high project failure rates for several reasons. Some ERP projects exceed the target budget and expected time. In some ERP project, employees were not ready to start using a new system (Forcht, Aldridge, & Shorter, 2007). ERP projects also struggled to attain the intended benefits due to their complexity caused by the integration of the different processes and dealing with a massive amount of data (Scholtz et al., 2016).

The complexity of the user interfaces (UIs) of ERP systems leads to various usability challenges. Although these challenges might have been pointed out in some studies, but they are still encountered.

As per the ISO/IEC 25010, usability is *"the capability of the software product to be understood, learned, operated, attractive to the user, and compliant to standards/guidelines, when used under specific conditions"* (ISO, 2011).

The International Data Corporation (IDC) in Denmark, Norway, and Sweden conducted a survey on enterprise application usability (Lykkegaard & Elbak, 2011). This survey had 300 participants from different organizations with annual revenues beyond 100 million US dollars. It turned out that usability is still a problem encountered by users working on ERP systems. The results of the survey indicated that 40% of the participants do not find enterprise applications easy to use.

Poor usability impacts the users' acceptance of a system (Seffah et al., 2006). Thus, usability affects the users' productivity and ability to work efficiently and effectively (Topi et al., 2005). For instance, Avon declared a \$125 million write-down for the costs of a SAP ERP implementation. In Avon's case even though the SAP ERP worked properly, it did not show a clear return on investment and some of the sales agents found it complex and left the company (Henschen, 2013).



### **1.3 Problem Definition**

There are a lot of usability problems to be dealt with when talking about enterprise resource planning systems. A study was conducted as part of this thesis to inquire about the problems that are faced by the users of the SAP Business One ERP system. The results showed that the two most impactful issues with respect to the users' work efficiency, support, and satisfaction are related to navigation and task support. The first problem is related to task, e.g., data entry, completion using multiple UIs instead of filling all the information in one place. Using multiple UIs for a single task requires more time and the memorization of numerous steps. The second problem is related to incomprehensible error messages that are displayed by ERP systems. These non-friendly messages stress out and sometimes mislead end-users.

### **1.4 Research Objectives**

This research aims to study the usability problems that users suffer from when using ERP systems. It studies the problem of having to use multiple UIs to accomplish a single task and seeks to show the effect of combining these UIs. Moreover, this research aims to design a system for multi-target validation that can be used by both developers and non-developers as an attempt to solve the problem of unclear error messages and to ensure that data is being validated properly.

### **1.5 Approach and Main Results**

A study was carried out to check the usability problems faced in ERP systems. The study showed that the following two problems were frequent: using multiple UIs to accomplish a single task and unclear error messages. Upon this result, an empirical study was executed to test the usability difference between accomplishing a task using multiple UI and a single adapted UI. The outcome showed that the single adapted UI saves time and users found it to be much better than the initial set of UIs. A sample of error messages from various SAP Business One UIs was analyzed. Based on this analysis, ten types of validation rules were suggested. This thesis presented the design of an approach for developing model-driven

multi-target UI validation. The latter is capable of executing on UIs, software classes, and databases.

## 1.6 Thesis Organization

**Chapter 2 – Background and Motivation:** This chapter gives background information on ERP systems including their importance in an enterprise, evolution and market growth, critical success factors, risks, and business productivity measures. It also presents previous work done on ERP usability problems.

**Chapter 3 – A Study that Investigates Usability Issues in the SAP ERP System:** This chapter presents the empirical study performed to examine the concrete usability issues faced in ERP system UIs.

**Chapter 4 – A Study that Assesses the Usability Impact of Merging Multiple UIs Based on the Task:** This chapter demonstrates the design and results of a study that evaluates the effect of merging UIs based on the task that an end-user has to perform. The effect is evaluated with respect to usability in terms of the end users' efficiency and satisfaction.

**Chapter 5 – An Approach for Developing Model-driven Multi-target Validation:** This chapter discusses the problem with error messages and the analysis of the error messages in an ERP system. Furthermore, it suggests different types of validation operations with the design of an approach for developing model-driven multi-target UI validation.

**Chapter 6 – Conclusion:** This chapter summarizes the main results and contributions of this thesis, and proposes future work.

**Appendix A – Questionnaires used in the Studies:** This appendix, as its name reveals, presents the questionnaires used in the two studies conducted in this thesis.

**Appendix B – Identified Error Validation Types:** This appendix presents the messages and validation types of the errors that were extracted from various SAP Business One UIs.

## **Chapter 2: Background and Motivation**

This chapter provides background information on ERP systems and presents the main motivation behind this thesis. An explanation is given on the importance of ERP systems in solving the lack of communication between the departments of an enterprise. This chapter also covers ERP evolution and market growth, critical success factors, risks, and business productivity measures. Finally, the previous work on ERP usability issues is presented.

### **2.1 Enterprise Departmentalization and Communication**

An organization divides work into several departments depending on the different activities they are assigned to perform. These departments usually include the following: sales, marketing, distribution, finance, purchasing, production, human resources, etc. This structuring and departmentalization by function increases efficiency and effectiveness. Members in the same team are qualified and specialized in a certain area; they usually come from a similar background and have common interests (Ahmed, 2017).

Every department collects and analyzes data with its own system and submits its own reports to the management based on its concerns without having any knowledge about the work of other departments. Sometimes, this may cause conflicts since departments could focus on their own functions and could have divergent objectives and independent goals when they are not working together for a common purpose. This lack of communication is one of the main challenges that face enterprises.

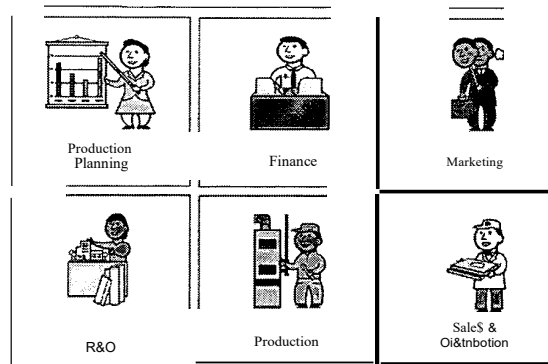


Figure 2-1 -lack of communication between departments of an organization (Leon, 2013)

Communication and coordination between departments are essential to achieve the organizational goals. For this reason, the data generated by an enterprise's departments should be managed by an integrated system in order to promote better communication and collaboration (Leon, 2013). Consequently, departments become knowledgeable of what others are doing and are able to plan what should be done to reach the company's goal.

And here comes the importance of ERP systems which integrate and automate business processes. Thus, all information is made accessible to the various departments in order to work as a single entity and accomplish the organizational goal. An ERP system can be very profitable for enterprises if it is implemented in the right way (Forcht et al., 2007).

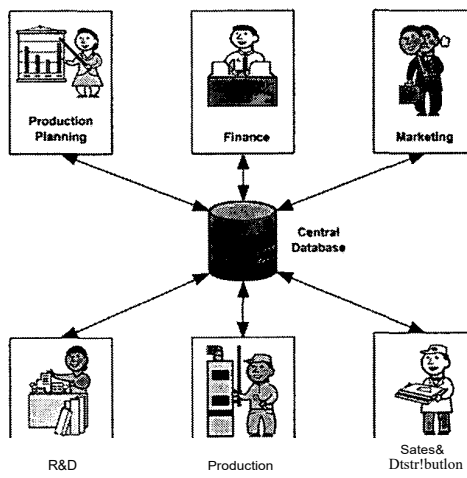


Figure 2-2 -Information shared between departments in an enterprise (Leon, 2013)

## 2.2 ERP System Evolution and Revenues

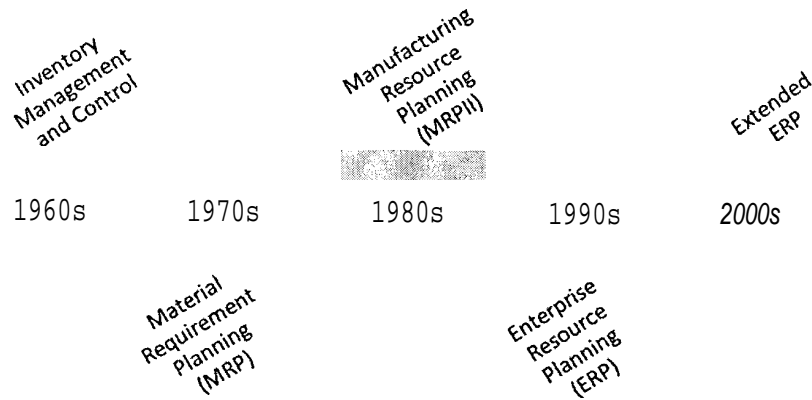


Figure 2-3 - ERP evolution timeline

The advancement of technology influenced the evolution of ERP systems. It all started in the 1960s when a simple system was needed to perform business calculations and keep track stock levels in warehouses (Wortmann, 1998). The developed system was called inventory management and control. It was responsible for maintaining the stock level by recognizing the missing, required, used, committed, and available inventory items. Besides, its activities involve setting targets, giving detailed reports about inventory status, and calculating future forecasts.

In 1970s, the material requirement planning (MRP) system was established (Leon, 2013). An MRP supports production decision making and the choice of raw materials needed for this production based on a bill of materials (BOM). An MRP also checks an item's availability in the stock and determines what components should be ordered to manufacture the desired products. It relies on transaction and production processing schedules to perform the production tasks and deliver the final product on time.

The ERP evolution continued during the 1980s and resulted in the creation of the manufacturing resource planning (MRP II) system. The latter provided additional

functionalities for managing the following: sales, distribution, finance, production planning, and inventory control to enhance the manufacturing processes.

ERP systems appeared on the market during the 1990s. These systems aim to facilitate the efficient and effective management of businesses. ERP systems integrate an enterprise's departments and automate all their activities including: sales, distribution, purchasing, accounting, finance, human resources, inventory, and production.

Finally, in the 2000s, with the increasing demand on additional functionalities and modules, extended ERPs were created (Hossain et al., 2001). Some example of the new modules included advanced planning and scheduling (APS), customer relationship management (CRM) and the supply chain management (SCM). These modules were developed as additions that extend the existing core activities.

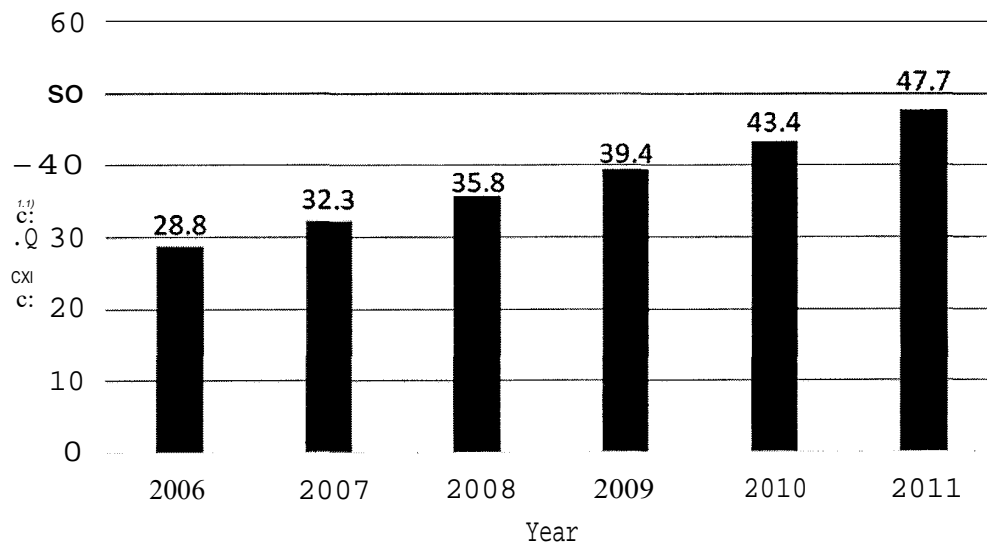


Figure 2-4 - Estimation of ERP application revenues between 2006 and 2011  
(Jacobson et al., 2007)

With the technological development and ERP evolution, the customers' demands increased. Hence, the ERP market grew throughout the years as shown in Figure 2-4. The revenues reached 28.8 billion US Dollars in 2006 and increased to 47.7 billion US Dollars in 2011 (Jacobson, Shepherd, D'Aquila, & Carter, 2007).

## **2.3 ERP System Critical Success Factors**

A number of articles studied the critical success factors (CSFs) of an ERP implementation in different countries and regions. CSFs were grouped into the following eleven categories: appropriate business and IT legacy systems, business plan and vision, business process reengineering, change management culture and program, communication, ERP teamwork and composition, monitoring and evaluation of performance, project champion, project management, software development testing and troubleshooting, and top management support (Nah, Zuckweiler, & Lau, 2003).

### **2.3.1 Appropriate Business and IT Legacy Systems**

The degree of technological and organizational changes needed depends on the complexity level of the legacy systems (Holland, Light, & Gibson, 1999). That is, the more complex the legacy systems are, the more changes are required. Dealing with complications coming from business and IT legacy systems is essential for a successful ERP implementation (Nah et al., 2003).

### **2.3.2 Business Plan and Vision**

It is important to have a clear business plan and VISION in order to direct an ERP implementation (Buckhout, Frey, & Nemeč Jr, 1999) since such projects usually go beyond their scheduled time (Nah et al., 2003). Moreover, the project mission, goals, and justification for investment should also be clearly stated beforehand based on the business needs (Ngai, Law, & Wat, 2008).

### **2.3.3 Business Process Reengineering**

The implemented ERP sometimes may not fit all the business requirements of a company. For this reason, customization is needed. However, it is better to reengineer business processes so that they become compatible with the ERP than to adjust the software (Sumner, 1999). Minimal customization is recommended to reduce the chance of errors and limit the increase in cost in order to profit from the new technology (Rosario, 2000).

### **2.3.4 Change Management Culture and Program**

When the necessity of change increases, the ERP implementation project will be approved by the management and stakeholders (Nah et al., 2003). Changes are managed on three different levels: people, organization, and culture (Rosario, 2000). Educating employees about the ERP systems and training them on processes directly related to their business are essential factors to accept and be prepared to use the new system (Ngai et al., 2008).

### **2.3.5 Communication**

Effective communication is required through all the stages of the company's ERP project; before, during, and after the implementation. The purpose, target, expectations, plans, and all the updates should be communicated to employees (Nah et al., 2003). And in return, the users provide their demands and feedback.

### **2.3.6 ERP Teamwork and Composition**

Every single operating department in a company, technical or business, participates in the ERP project. Teamwork and cooperation between the experts of the enterprise, end users, vendors, implementers and consultants are vital (Nah et al., 2003). This interaction between the team members, outsiders and insiders, contributes to the project's success (Haines & Goodhue, 2000).

### **2.3.7 Monitoring and Evaluation of Performance**

Goals should be set and supervised to check the advancement in the project (Nah et al., 2003). Furthermore, end-users' reactions should be considered while monitoring and evaluating an ERP system's performance and an implementation's progress (Ngai et al., 2008).

### **2.3.8 Project Champion**

Commitment and perseverance are important elements that contribute to the success of an ERP project, since such projects need a lot of effort and long hours of work. For this



reason, a project champion is required. The latter is responsible to ensure acceptance of the system by users, resolve issues, and raise project team members' morale (Nah et al., 2003).

### **2.3.9 Project Management**

The project management task, handled by a particular person or by a group, is of high importance since it impacts the success of the ERP implementation (Nah et al., 2003). Project management is responsible to keep track of the schedule, plan, scope, targets, and budget. It makes sure that deliverables at the end of each implementation stage are provided on time as well as all sort of faced problems and proposed changes are managed.

### **2.3.10 Software Development, Testing and Troubleshooting**

It is not simple to implement a software, especially an ERP. Development is needed to integrate an enterprises' legacy system and its new ERP system (Ngai et al., 2008). Furthermore, to check that the system is working properly and that errors and conflicts are fixed, testing and troubleshooting are essential. This requires the team members responsible for the ERP implementation within an organization to work together with their consultants (Holland & Light, 1999).

### **2.3.11 Top Management Support**

To get a successful ERP implementation, the top management's approval and support are needed along with setting this project as a priority (Nah et al., 2003). Then, the top management should assign resources for the implementation such as time, budget, and employees. The more committed the top management is, the more the employees become committed which impacts the project's success (Bingi, Sharma, & Godla, 1999).

## **2.4 ERP Risk Categories**

ERP implementations are complex and require a lot of resources. Such projects consume a lot of time and money, sometimes beyond what is expected. ERP implementations also require qualified employees to keep track of the steps to be followed in order to reach a successful implementation. They need to make sure that the new ERP system meets their

company's business requirements. The following three categories of fundamental risk factors are linked to the implementation process: people (69%), process (18%), and technological (13%). Each of these factors constitutes a percentage of the total impact as shown in Figure 2-5 (Leon, 2008).

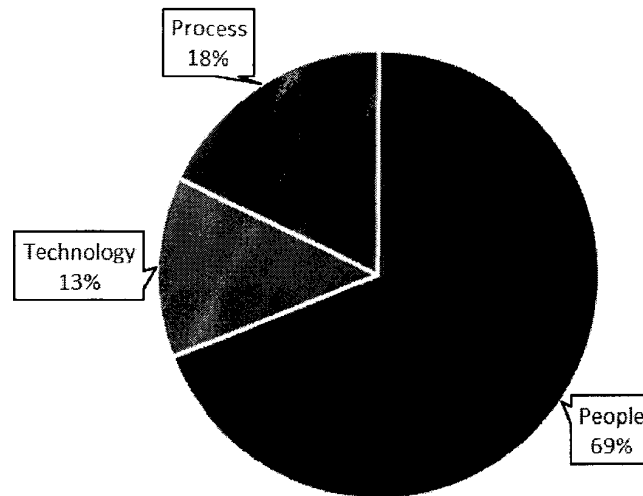


Figure 2-5- Categories of Risks in ERP Systems

### 2.4.1 People issues

It is normal that people sometimes oppose change. Changes in the organization affect people's daily routine thereby increasing their sense of fear; fear of losing control, stability, security, and predictability they already have in their lives (Adenle, 2011). Employees might think that since they are able to perform all what is required using the old system, there is no need to change it (Stark & Flaherty, 2011). Users should be prepared for change prior to the implementation phase. The advantages and significance of using the new ERP in the business process should be made clear to them. It is important to persuade them to cooperate in order to have a successful implementation. Change management, internal staff adequacy, project team, training, employee relocation and re-training, staffing (including turnover), top management support, consultants, discipline, and resistance to change are the principal people issues (Leon, 2008).

### 2.4.1.1 Change management

When an ERP system is implemented, some tasks that end users are used to perform will be automated and others will no longer be needed. On the other hand, new tasks are also introduced. Changes that will be applied on the business functions and processes should be managed appropriately in order to minimize the risk of failure. In addition, the project's mission statement along with the project requirements and the change control procedures should be clearly stated to all parties to overcome any problem that might be encountered.

### 2.4.1.2 Internal staff adequacy

An internal team of skilled employees from inside the company is usually involved in the ERP implementation process. The implementation also involves external stakeholders who can be vendors, consultants, and representatives of the new system. The implementation cost increases when there is a lack of skilled employees since in this case, the company is obliged to employ more consultants.

**Project team:** The group assigned to work on this complex implementation should be based on devoted and qualified employees with good teamwork and communication skills. Otherwise, the management would be putting the success of the implementation as risk.

**Training:** It is essential for end-users to learn how to use an ERP system. Therefore, the lack of sufficient training, especially by the end of the implementation, e.g., because the project cost surpassed the planned budget, results in ERP implementation failure.

**Employee relocation and re-training:** With the change in tasks and processes as described earlier in the change management part, the job descriptions are changed. It would cause an issue if the organization does not handle this change in a friendly way with the employees explaining to them the importance and benefits of what is being accomplished for their own sake and for the organization.

**Staffing (including turnover):** It is necessary to have skilled employees through all the phases in the transition, implementation of the ERP system, and post-implementation phases. ERP implementation is a complex process and requires a lot of work and effort. Therefore, the personnel should be rewarded and kept satisfied to avoid turnover issues with qualified employees. Employee turnover could delay a project's schedule and increase

its cost. Thus, a backup plan should be ready if a trained employee left the company during these critical phases.

**Top management support:** In order to support the implementation process, top management needs to be totally convinced that implementing an ERP system is for the good of the enterprise. One way of supporting the implementation is by managing employee resistance. Moreover, management could provide the project team with all the required resources. Failure is the outcome of an ERP implementation that is not fully encouraged and fortified by the top management.

**Consultants:** They are the experts that take care of the implementation and configuration of the ERP system to meet the business needs of an enterprise. If the consultants do not know what the enterprise does and what it needs, there will be a huge problem. To be on the safe side, it would be useful to assign an employee to make consultants more familiar with the work environment and mediate the relationship with the implementation team.

**Discipline:** Employees and managers are required to learn how to use the system. Managers need to observe and check the phases that the new ERP system is undergoing. They also have to know how to make accurate decisions by interpreting the data produced by the ERP system. Employees are required to learn how to use the ERP system to properly carry out their tasks.

**Resistance to change:** Once the ERP team, along with the management, informs the employees of the ERP benefits and how it will facilitate their work, the bias they might have in mind against ERPs will no longer seem justifiable and consequently so will the resistance to change.

### **2.4.2 Process risks**

Managing business processes contributes to the success of ERP implementations. This is essential since with ERP systems some tasks and processes will be removed and new processes will be generated to make work more effective and efficient. The basic elements of process risks are program management, business process reengineering, stage transition, and benefit realization.

#### **2.4.2.1 Program management**

ERP systems are made of multiple modules enabling them to cover a wide range of functions. All the activities require available real-time information, but not all the information can be obtained from the ERP. A part of this information is present in the ERP system and the other is related to program management information which is vital for the program execution and cannot be saved in traditional ERPs. Thus, this information gap caused by the absence of a link between the two types of functions can be solved with program execution and management applications. They help in bridging the gap to supply up-to-date information for better decision-making.

#### **2.4.2.2 Business process reengineering**

The use of information technology accompanied by training, changes in jobs, organizational structure, management systems, and so on, results in dramatic changes that impact the business processes. Hence, these changes might produce either a great success or a huge failure.

#### **2.4.2.3 Stage transition**

A lot of people work together on the ERP implementation, but at the go-live stage new roles are assigned. After the ERP starts operating, every person involved in the project will know their responsibilities to avoid conflicts related to decision making responsibilities.

#### **2.4.2.4 Benefit realization**

Following the ERP implementation, the operational phase begins. This phase should involve planning, use of the system properly with all its features, management full support, personnel's training and participation. Otherwise, there will not be any remarkable increase in benefits and profit.

### **2.4.3 Technological risks**

The organization must progress at the same rate as the technological improvements to stay competitive in its market. Software functionality, technological obsolescence, application

portfolio management, enhancements and upgrades are a few of the encountered technological issues.

#### **2.4.3.1 Software functionality**

Organizations might not use all the features that are supported by their ERP system. These unused features vary depending on the organization and its requirements. The consultants are required to communicate with the management to check which features should be disabled in order to avoid end-user confusion and a negative impact on usability. In addition to the usability problem, it takes time and effort to install all the ERP system's features thereby resulting in a cost increase. Additional features can be installed separately if they are required after the implementation.

#### **2.4.3.2 Technological obsolescence**

With time, all existing technologies will become outdated because of the fast and continuous technological development. For that reason, companies need to carefully choose the vendor and the system that are best suitable and can keep up with the pace of advancing technology. Otherwise their software system would not become obsolete within a short period of time.

#### **2.4.3.3 Application portfolio management**

Some IT organizations take care of different systems, applications, and tools in addition to several kinds of the same system for example various ERP systems. They manage, maintain, and upgrade these systems thereby increasing the expenses and the number of redundant complications. To handle this issue, an application portfolio is used to manage the organization's applications and decide based on metrics whether or not an application should be kept, updated, withdrawn, or replaced.

#### **2.4.3.4 Enhancements and upgrades**

Updating modules and features as well as upgrading the version of an ERP system is always needed to have maximum efficiency. Updates are provided by the vendor when available. However, updates are a risk for the organization since it relies on the vendor,

especially if the vendor stops providing support for the system. To be on the safe side, it would be better if an organization signs a contract with the vendor.

## 2.5 ERP Business Productivity Measures

Software applications like ERP systems affect the end users' productivity while performing their daily tasks. This depends on the applications' ease of use and the extent to which they help in meeting users' needs. A business productivity measurement framework was developed to examine the impact of such applications on business performance and more precisely the end users' productivity (Keystone Strategy, 2007). As summarized in Figure 2-6, this framework is based on the following six components: usability, collaboration, business insight, flexibility, transactional efficiency, familiarity.

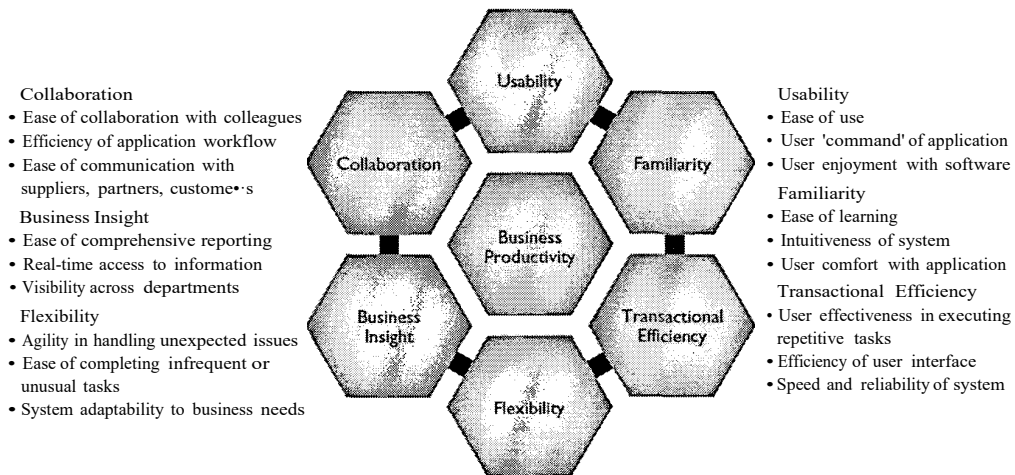


Figure 2-6 -Business Productivity Measurement Framework (Keystone Strategy, 2007)

### 2.5.1 Usability

It is the factor that indicates the degree to which a system is easy to use in order to perform tasks with effectiveness, efficiency, and satisfaction. High usability encourages users to use the software more often. When it is easy to navigate through an application, find required data, and explore new functionalities, users will be pleased and their performance and productivity will increase.

### **2.5.2 Collaboration**

It is important to collaborate and communicate with other individuals to perform, review and contribute to a certain work together using an application. These individuals can be a group of colleagues from the same or different departments within the company, or from outside the company like customers and suppliers. Good collaboration organized by the software within an organization and with outsiders, improves end users' productivity and business environment.

### **2.5.3 Business Insight**

This dimension determines how fast end-users can access real-time accurate and appropriate data, even if that data was added to the system by different departments. In addition, business insight measures how easy it is to get an understandable and detailed report with all the required information from this system to help the management take quick and right business decisions. When users feel that they are getting the outcome they need in a short period of time and their effort is not wasted, they will enjoy using a software that facilitates their work and reduces the time spent on reporting.

### **2.5.4 Flexibility**

This factor indicates to which extent a software application can easily respond and adapt to changes like user requirements and new business needs. It indicates easy and agility of addressing encountered problems and unfamiliar tasks. Flexibility is taken into consideration when choosing a software application, knowing how often business requirements changes occur and affect business productivity.

### **2.5.5 Transactional Efficiency**

It is the aspect that rates how simple, consistent, and efficient it is to achieve common and recurrent tasks using the system's interface. High transactional efficiency decreases the end users' resistance towards the software they are using and increases their productivity.



### 2.5.6 Familiarity

This property indicates how much end-users feel that a software is friendly and easy to learn and use. When familiarity is high, end-users do not take much time for training to become familiar with an application and then experts in using it. Familiar systems offer friendly functions, interfaces, and features that inspire end users to discover its capabilities and search for new functions. Users' comfort and satisfaction augment their performance and as a result the business productivity.

## 2.6 Previous Work on Usability Problems in ERP Systems

Many studies were conducted to investigate the usability problems facing ERP users. The question asked by these studies include "why users find ERP systems difficult to learn and use" and "what are the features that bother users while performing their regular tasks". The results of these studies are summarized in this section.

Common usability heuristics that are usually used to evaluate systems include Nielsen's ten heuristics and Shneiderman's eight golden rules for UI design. However, there is no typical way to study ERP usability in particular. Most of the usability issues in ERPs were arranged in five categories shown in Figure 2-7 (Singh & Wesson, 2009).

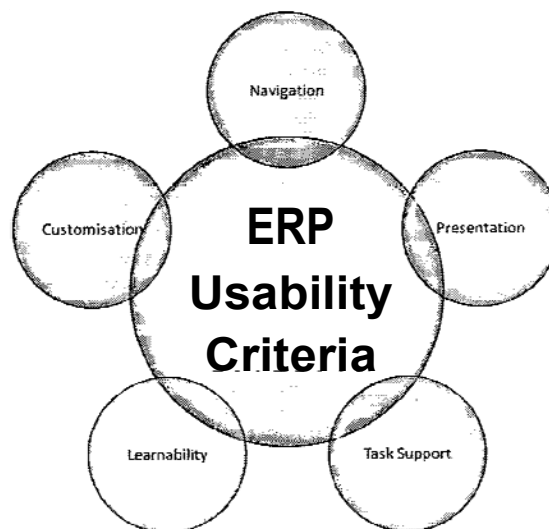


Figure 2-7 - ERP Usability Criteria (Singh & Wesson, 2009)

### 2.6.1 Navigation

Navigation is related to the end-user's ability to effectively find and access relevant information and functionalities using menus, options, and reports. A large number of ERP systems suffer from navigation problem. The following UI navigation problems were discussed in existing research papers as indicated in Table 2.1:

1. *Invisible system status*: the users cannot indicate, at a certain moment, the exact status of the system. Also, it is hard for users to know their present place within the system and the functions that can be executed at this stage.
2. *Problem with identifying and accessing information and functions*: it is complicated for users to discover the location of the needed information and to find the right functions while looking through the menus and numerous levels of submenus.
3. *Insufficient guidance to perform the next steps*: the data entry procedures along with the sequence of steps that should be conducted to finish a task are not evident for users.
4. *Problem with search option*: in some cases, there is no search option. However, in others, this feature is found but is instable and does not work properly.
5. *Difficulty in knowing the difference between certain options*: there is a kind of uncertainty and ignorance about the difference between the command and search boxes and likewise between the lookup and search buttons.
6. *Need for cheat sheets*: the complexity of Uis imposes the use of work instructions created for the company's employees.
7. *Problem while working with lists*: fundamental navigation and selection inside lists are not easily perceived and inconsistent.
8. *Entering redundant data repeatedly into different Uis*: the users are obliged to add the same data manually over and over again into different Uis instead of having the system fill them automatically. This increases the probability of errors and discrepancies in the system, in addition to the waste of time.
9. *No support for shortcuts for often used commands*: system is not flexible; creating shortcuts for regularly carried out commands is not supported.

**Table 2.1 -Usability problems related to navigation**

<b>Usability Problems</b>	<b>Papers</b>
1. Invisible system status	(Singh & Wesson, 2009), (Oja & Lucas, 2010), (Babaian, Lucas, Xu, & Topi, 2010)
2. Problem with identifying and accessing information and functions	(Topi et al., 2005), (Singh & Wesson, 2009), (Oja & Lucas, 2010), (Scholtz et al., 2010), (Babaian et al., 2016), (Lambeck et al., 2014)
3. Insufficient guidance to perform the next steps	(Singh & Wesson, 2009), (Babaian et al., 2010), (Oja & Lucas, 2010)
4. Problem with search option	(Singh & Wesson, 2009), (Oja & Lucas, 2010)
5. Difficulty in knowing the difference between certain options	(Scholtz et al., 2010)
6. Need for cheat sheets	(Topi et al., 2005), (Babaian et al., 2010)
7. Problem while working with lists	(Oja & Lucas, 2010)
8. Entering redundant data repeatedly into different UIs	(Topi et al., 2005), (Scholtz, Calitz, & Cilliers, 2013)
9. No support for shortcuts for often used commands	(Singh & Wesson, 2009)

## 2.6.2 Presentation

Presentation is another serious usability issue pointed out in the studies. This heuristic aims to discover the complexity level of the screens and the output for users to get proper and understandable information. The main problems encountered are shown in Table 2.2 and described as follows:

1. *Problem with data entry basic rules:* elementary regulations such as the format, restrictions, optional and mandatory fields are not clearly specified when users are doing data entry.
2. *Complexity of visual layout:* the visual design of the screens is complex since UIs are overloaded with information and screens are cluttered with fields. Furthermore, the visual design labels are unclear and the items' grouping are illogical.

3. *Problem with retrieved output*: the output from the system is difficult to comprehend and extract. Also, in some cases, there is a need to use a supplementary software like Microsoft Excel spreadsheet to fulfill the reporting requirements that are not applicable in the ERP system.

**Table 2.2 - Usability problems related to presentation**

Usability Problems	Papers
1. Problem with data entry basic rules	(Singh & Wesson, 2009), (Oja & Lucas, 2011)
2. Complexity of visual layout	(Singh & Wesson, 2009), (Oja & Lucas, 2011), (Scholtz et al., 2010), (Scholtz et al., 2013)
3. Problem with retrieved output	(Topi et al., 2005), (Singh & Wesson, 2009)

### 2.6.3 Task Support

Task support is a criterion that expects to determine the accuracy of the alignment between the real world and the system and if users are offered the effective help and task support to complete their tasks efficiently. The mismatching between the system and the real world increases the complexity of using the ERP system causing intolerance to the system. The reported issues related to this aspect are shown in Table 2.3 and described as follows:

1. *Problem with the support in error situations*: feedback and error messages are sometimes confusing or not suitable in current circumstances. Trying to search for the origin of the issue wasted the users' time. The messages may be too general or it might not show the severity of the error or clearly state its cause. The messages might also not suggest a solution to the error to help users solve the problem. Besides, specific messages are displayed in diverse contexts increasing user's confusion.
2. *Lack of basic controls for error prevention*: some necessary errors that should prohibit users from saving wrong data entries are not displayed. Moreover, fields do not have default values to facilitate the users' job.
3. *Location of notifications*: users can fail to notice warnings, information messages, and field prompts since they are shown at the bottom of the screen.

4. *Useless help*: system is complex for novices; the information provided by the help in the system is limited, unclear, and general. Furthermore, the documentation does not explain the precise tasks at hand.
5. *Absence of system support*: missing system support that helps in understanding and successfully completing one's tasks.
6. *Mismatch of terminology between system and real world*: the used terms within the system are broad and inconsistent with that known by users.
7. *No undo option*: once an action is done on the system, it cannot be reversed.

**Table 2.3 - Usability problems related to task support**

<b>Usability Problems</b>	<b>Papers</b>
1. Problem with the support in error situations	(Topi et al., 2005), (Singh & Wesson, 2009), (Oja & Lucas, 2011), (Babaian et al., 2010)
2. Lack of basic controls for error prevention	(Topi et al., 2005), (Singh & Wesson, 2009)
3. Location of notifications	(Singh & Wesson, 2009), (Oja & Lucas, 2011)
4. Useless help	(Singh & Wesson, 2009), (Oja & Lucas, 2011), (Babaian et al., 2010)
5. Absence of system support	(Topi et al., 2005), (Singh & Wesson, 2009)
6. Mismatch of terminology between system and real world	(Topi et al., 2005), (Singh & Wesson, 2009), (Babaian et al., 2010)
7. No undo option	(Singh & Wesson, 2009)

### 2.6.4 Learnability

Learnability was proposed as one of the usability criteria since ERPs are considered complex and difficult to learn and use. As a consequence, it plans to establish the amount of effort needed in order to learn about the system and how to use it in an efficient way. The learnability problems identified by previous studies are shown in Table 2.4 and described as follows:

1. *Problem with learning and remembering features*: it takes a lot of time for users to learn and memorize the needed features of the system. They also shall recall every single time the series of required operations that must be performed in a special order to be able to complete a business process.
2. *Problem with understanding functionalities*: it is hard for users to interpret the reason and work of some functions and their sequence in the system to accomplish a particular task.

**Table 2.4 - Usability problems related to learnability**

Usability Problems	Papers
1. Problem with learning and remembering features	(Topi et al., 2005), (Singh & Wesson, 2009), (Babaian et al., 2010)
2. Problem with understanding functionalities	(Topi et al., 2005), (Singh & Wesson, 2009), (Oja & Lucas, 2011)

### 2.6.5 Customization

Customization is assessed based on the ability of customizing the system easily to meet the enterprise requirements, support its business processes, and meet the users' individual needs. Customization is one of the characteristics of ERP systems. However, these systems face the customization problems shown in Table 2.5 and described as follows:

1. *Difficulty in altering user's settings*: it is difficult to modify the users' settings depending on their own wishes.
2. *Finite customization allowed*: little and restricted customization is allowed in such systems.

**Table 2.5 - Usability problems related to customization**

Usability Problems	Papers
1. Difficulty in altering user's settings	(Oja & Lucas, 2011)
2. Finite customization allowed	(Singh & Wesson, 2009)

## **Chapter 3: A Study that Investigates Usability Issues in the SAP ERP System**

Empirical evaluation of UIs is essential to assess their usability using both qualitative and quantitative data collection and analysis methods. For this reason, a study was conducted to check the usability problems encountered in real-life ERP system UIs.

### **3.1 Study Design**

This study was performed in an Internet Service Provider (ISP) enterprise that uses the SAP Business One ERP system. SAP Business One is a business management application designed for small and medium -sized enterprises (SAP, 2017). It supports all the functions required to complete the business process. These functions include financials, sales, opportunities, inventory, purchasing, human resources, production, and banking. All these processes are automated and integrated in SAP Business One. This helps the end users to access real time information immediately and create reports with the needed data from different departments of the business.

The participants were asked to describe and rate the severity of the usability problems that they frequently face during their daily use of SAP Business One. The results showed that one of the most common problems was the frequent need to use multiple UIs in order to make a single document entry. Filling data in multiple windows for a single activity made the system complex to use. Another familiar problem is the unclear error messages and codes that users receive in error situations. These messages confuse users instead of helping them to find their mistakes and correct them. These two main issues reported by users are studied in this thesis; Figure 3-1 shows the thesis design.

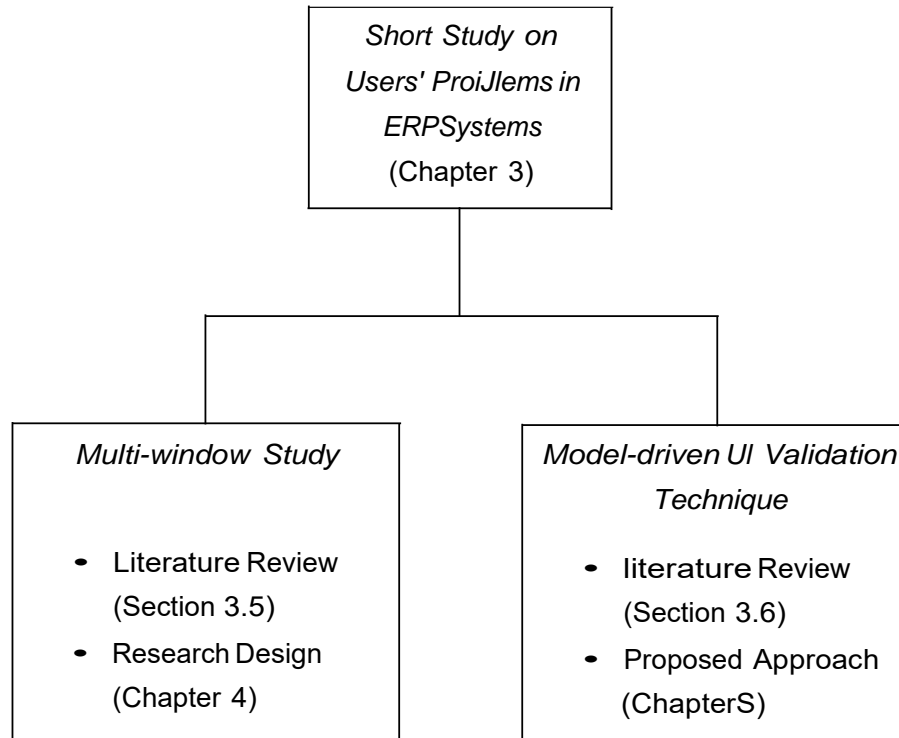


Figure 3-1 -Thesis diagram

### 3.1 Participants

Twenty participants from the ISP employees volunteered to take part in this study. Ten of them were randomly picked from the sales department and another ten from the accounting department. The study took an average of 10 minutes per participant. The demographic information of the participants varied in terms of gender, age group, enterprise application experience, hours of work per day, and computer skills (self-evaluated). The participants' background information is illustrated for the accounting and sales personnel in Figure 3-2 and Figure 3-3 respectively. This information is shown separately per group because each group of participants assessed a different UI in the multi-window study.



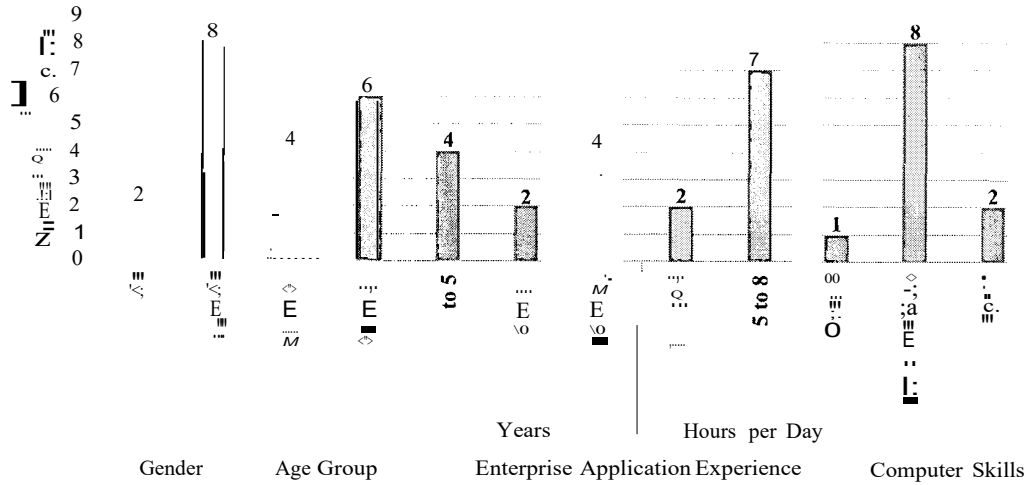


Figure 3-2 - Demographic information of participants from accounting

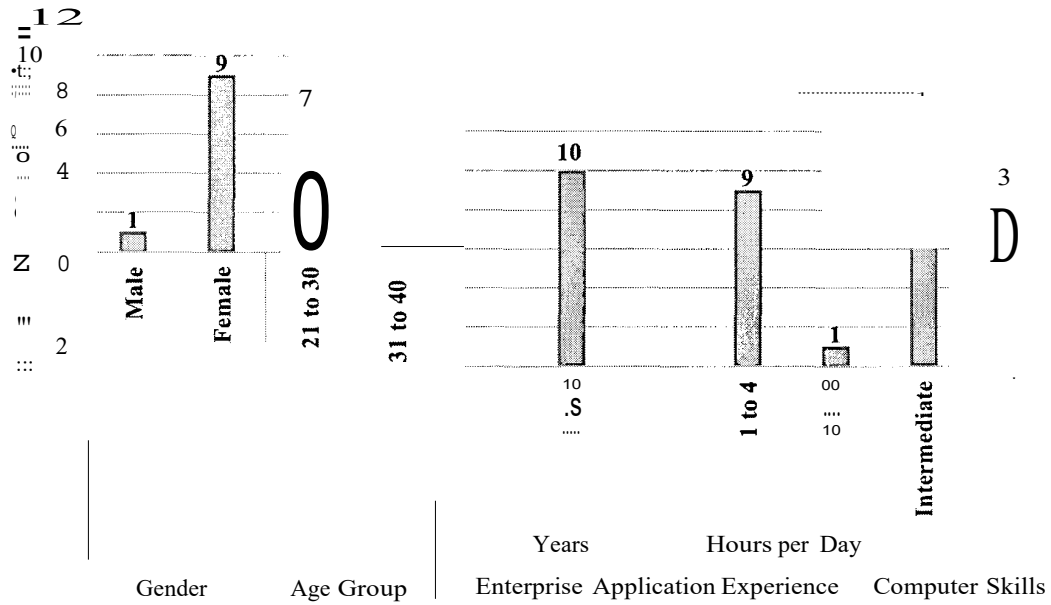


Figure 3-3 - Demographic information of participants from sales

### 3.2 Study Results- SAP Usability Problems

This section reports the recorded usability problems that participants suffer from while doing their tasks. To check SAP usability problems, participants were given the chance to express the main usability problems they encounter in their daily use of SAP and were requested to rank the severity of each problem. The ranking was done on a scale of 1 to 5,

where 1 indicates that a problem is extremely severe and 5 indicates that it is somewhat severe. The results of this ranking are presented in Table 3.1. This table shows the problems identified by the participants in addition to the mean, median, standard deviation, and standard error of the participants' severity ranking.

To elaborate more on the problems reported in Table 3.1, it is possible to say that ERP users suffer from many usability problems that are related to different usability criteria. The problems identified in this study were categorized under the following usability criteria: navigation, task support, learnability, presentation, and customization. These criteria were previously explained in the literature review (section 2.6).

**Table 3.1 -SAP Usability Problems Reported by Participants**

	Reported Problems	Number of Participants	Problem Severity (1=Extremely Severe to 5=Somewhat Severe)			
			Mean	Median	SO	SE
Navigation	Many windows should be opened and filled to be able to complete the required document	8	3.5	3.5	0.53	0.19
	To add a document, one must pass through many steps and pop-up messages					
	Several menu options should be selected to reach a required query		4	4		
	Some windows cannot be minimized to work with other windows	2	2	2	0	0
Navigation & Learnability	PrJ le \, itb searc f t, and so					0.33
Task Support	Upc					0.7
	Sometimes no errors are displayed to prevent wrong entry	3	2.33	3	1.15	0.67
	No undo option	2	2.5	2.5	2.12	1.5
Task Support & Learnability	Help support and documentation are needed		3	3		
Presentation	Screen\$,contn a lot		3.4	4	0.89	0.4
Customization	Inability to customize the display of all window fields upon one's wishes	2	3.5	3.5	0.71	0.5
Customization & Presentation	No option to select what should be displayed when printing a document upon one's wishes		4	4		

*Navigation* problems are common in ERP systems such as SAP. These problems include going through several phases to reach a target. Hence, to complete a sales application or an accounting transaction, one must perform data entry in several windows. For example, to issue a credit note for a customer, the phone number and the amount are obtained from two separate windows, namely business partner master data and invoice. Additionally, the statement of account is checked beforehand in another window. Another example is that of manual reconciliation. When a remark is needed to be added for the reconciliation, the accountant should go to the customer's journal voucher window to add it. Moreover, when adding a document such as an incoming payment, the end user receives several pop-up messages before saving the payment. End users are required to navigate through four levels of a nested menu in order to get to the desired query. Furthermore, in some cases end users cannot open more than one instance of a UI. For example, assume that an end user is working on a statement of account. It is not possible for this end user to open another instance of the statement of account UI, until the first one is closed. However, opening multiple instances of a UI is sometimes required, for example, when customers call to make inquiries while the end user is working. Adding to that, the search, filter, and sort options are sometimes unavailable. In other cases, these options yield wrong results such as a long list of records that does not match the intended selection. In other cases, end user gets confused because it is hard to locate these options and learn how they work.

Concerning *task support*, SAP sometimes issues error codes and messages that are technical and confusing to end users. Examples of these technical messages include: "Invalid series next number [Message 3535-3]" and "Internal error (-2014) occurred [131-183]". These error examples do not fully describe the error in layman terms that allow the end users to figure out a solution. In addition, sometimes when critical wrong data entries require an error to prevent users from saving their work, no error is displayed. For instance, no error is displayed while saving invoices with wrong amounts, such as manually added unit prices that are predefined at the item level and should not be changed, or when negative discounts are entered. Also, no warning is displayed while integrating a file to prevent the import if there is a wrong entry or an empty invoice in case of recurring billing file. Another issue the participants faced during their daily work is that there is no undo option to reverse a wrong data entry or modify. This forces them to undergo a long process

of cancelling or closing the document. For instance, to adjust a wrong invoice, one should issue a credit memo and then issue a correct invoice again. There is a problem as well with the help support and documentation which are not available to help users learn and complete certain tasks successfully for example when filling windows with multiple options like the trial balance report grouped by segment.

As for the *presentat;on* criteria, Uis are in many cases cluttered with fields making it difficult for end users to locate the mandatory fields.

Regarding *customhat;on*, users are unable to customize windows upon their wishes. For example, users cannot set different colors to distinguish similar windows like the A/R invoice and A/P invoice. They are also incapable of adjusting the field size in the basic SAP windows, like the remark field at the bottom of the payment which is small and does not show all its content. And there is no option to select what should be displayed on the invoice for certain customers like the option to hide items and subtotals and just display the final total amount. It should be noted that the highlighted problems in Table 3.1 that have the larger number of participants are mutual to the two groups of users: the sales and the accounting users.

### 3.3 Threats to Validity

The study only involved two groups each with ten participants who worked on a single ERP system, namely SAP Business One. A larger number of participants working on different ERPs could provide richer and more generalizable results.

However, this study was done on SAP since it has one of the largest market shares (Panorama Consulting Solutions, 2016). Hence, it is possible to say that the results describe significant usability issues. Therefore, it would be useful to solve at least some of the usability problems exposed in this study, in order to improve the daily work of ERP end users.

### 3.4 Existing Work on UI Composition

Various techniques have been used to build UIs that facilitate the users' tasks. One of these procedures is "Visual design by de(composition)", which was developed for graphical UIs that are hard to adapt to fit a new context of use (Lepreux, Vanderdonckt, & Michotte, 2007). This method offers a sequence of operations to decompose or compose a UI in another context of use like a different platform or for another user. An existing user interface can be decomposed into several components that can be reused in upcoming designs. On the other hand, this method allows assembling one or more elements to compose a new UI.

Another concept is "graceful degradation" that was introduced for designing UIs to work with multiplatform systems (Florins & Vanderdonckt, 2004). Every platform is characterized by certain capabilities such as screen size and resolution. However, end-users assume that a system operates in the same way regardless of the used device or platform. For this reason, the graceful degradation approach was initiated to ensure a smooth transition and continuity between the different system platforms. This design method suggests a series of transformational rules that are executed with a priority ordering between them to guarantee the continuity between the system versions.

### 3.5 Existing Work on UI Validation

It is highly recommended to validate the input of users to prohibit saving improper data. Therefore, data validation rules are set to control what data the user is capable of entering in a certain field. Form validation can be applied in diverse ways some of which are: making sure that essential fields are not left empty, confirming that certain fields like numerical fields, date, phone number, user id, and email have the right format, validating two related fields, and checking if an option is selected from a list of checkboxes or radio buttons (Harwani, 2010).

Validation can be initiated automatically upon submission or when a change in a form field. However, it is easier for users to tolerate adjusting errors separately than treating a bulk of errors at the end at the submission time (Resig, Ferguson, & Paxton, 2015).

The form validation may be done on two levels: the client-side and the server-side. They are both used to check whether the entered data is accurate or needs to be fixed. Validation at the client-side gives an immediate response on the form before sending the data to the server. This type of validation is done by languages such as JavaScript using libraries such as jQuery (Lengstorf & Wald, 2016). Nevertheless, the application would become more secure if server-side validation was added. It works sometimes like a second line of defense against untrusted malicious attacks that bypass the client-side validation for instance when the user turns off JavaScript. Web applications for instance, are vulnerable to attacks known as script injections, dataflow attacks, or input validation attacks (Li & Xue, 2014).

Several approaches tackled the subject of form validation as an attempt to solve the wrong data input issue. One of the presented techniques depends on an abstraction termed "tope" independent of any software application (Scaffidi, Myers, & Shaw, 2008). Its validation code is reusable in other platforms and software. This technique helps in detecting doubtful input values and data with several formats. Input values that are questionable if they are valid or not, are found and sent to be double-checked by the user or by another program. And the data appearing in different formats are transformed to have a single consistent format understandable by the application. This solution enhances validation accuracy and data cleansing later on.

A server-side model-driven form validation approach was suggested for web applications. It consists of setting restrictions for the entered data in the model via the Unified Modelling Language (UML) and the Object Constraint Language (OCL) (Escott, Strooper, King, & Hayes, 2012). This approach determined three types of validation: single element, multiple element, and entity association. The *single element* validation is related to checking an HTML element using UML. The *multiple element* validation is achieved using OCL when an element's value impacts another element. The *entity association* validation addresses the class relationships in the domain model.

An alternative approach for the validation of web application models relies on using a domain-specific language called WebDSL (Groenewegen & Visser, 2013). This approach aims to integrate the data validation rules with the UI models of a web application. It categorized data validation into four main kinds: *value well-formedness* checks if the input

abides by the acceptable syntax, *data invariants* are constraints specified in the domain model, *input assertions* are defined for the entered data that are not directly linked to the domain model, and *action assertion* are validations done during the execution of an action.

### 3.6 Research Motivation

The navigation problem related to the use of multiple windows was addressed in several research papers. Basically, technical work was done on this subject without empirical evidence. For this reason, it is useful to do a short experiment to test if decreasing the number of UIs used to complete a task would be significantly more usable than using multiple UIs. On the other hand, to deal with the issue of unclear error codes and messages, a validation technique can be developed to allow the definition of dynamic validation rules more easily. Currently in SAP Business One, for example, validation rules are defined in long stored procedures written in the Structured Query Language (SQL). These procedures are hard to maintain. Simplifying the way these rules are defined might allow various stakeholders to work on rule definition. Furthermore, it would be interesting if the validation rules were applicable to multiple levels including: domain model, database, and user interface at the same time. There is no earlier work done on the developing validation rules that operate on all these three levels. Moreover, the proposed method is dynamic and more organized than previously suggested approaches and can be used by both developers and non-developers. The latter could be employees using ERP systems within an enterprise.

## **Chapter 4: A Study that Assesses the Usability Impact of Merging Multiple UIs Based on the Task**

The multi-window study assesses how usability is impacted by merging multiple UIs into one based on the task at hand. This chapter explains the designs of this study and presents its outcome. The results of the study showed that the adapted UIs exhibited a significant improvement in the efficiency and satisfaction of end users.

### **4.1 Study Design**

In this study, initial and adapted versions of two SAP Business One UIs (sales opportunity and invoice) were compared. Each group of participants worked with the UI that fits its job description. Hence, the sales opportunity UI was assessed by the sales personnel and the invoice UI was assessed by the accountants. They were asked to add a new record in the initial and adapted versions of their respective UIs. Once the documents were added and the participants experienced the two UIs, they were asked to answer the evaluation questions. The study took around 15 minutes per participant.

In order to complete an activity in the initial versions of the UIs, end users usually need to go through several windows to add the required data. On the other hand, the adapted UI versions consolidate all the necessary UI elements in one window. Hence, this study checks if the adapted UIs will significantly improve usability and thereby facilitate the end users' work.

The tools used to adapt the UIs in SAP Business One were the Form Settings and the User-Defined Fields Management as shown in Figure 4-1 and Figure 4-2 respectively. The Form Settings function was used to adjust the columns in the adapted UIs. This feature helps in selecting the fields that need to be visible in a specific window and in rearranging their order. However, the User-Defined Fields Management customization tool was utilized to create the additional fields required in the adapted UIs.



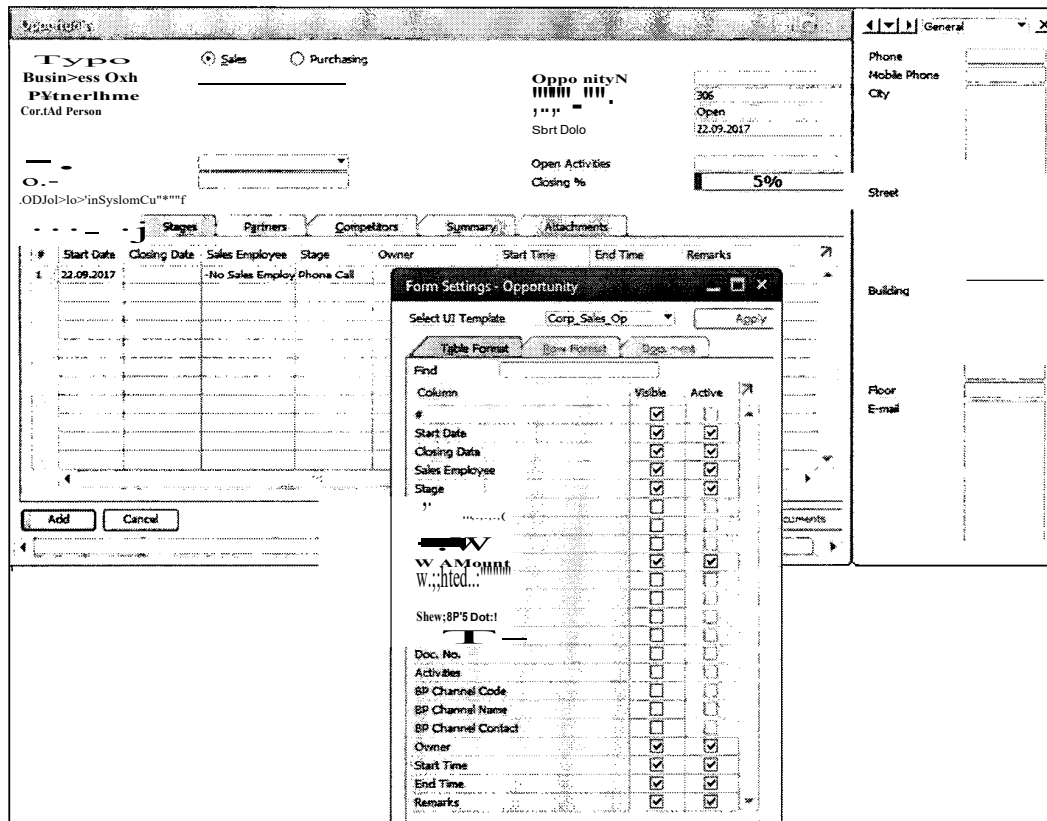


Figure 4-1 -Organizing fields using Form Settings

The adapted UIs decreased the number of windows while maintaining all the essential and mandatory fields. For instance, to add a sales opportunity in the initial SAP UIs, the sales agent is required to insert data in three separate windows: business partner master data, opportunity, and activity windows. These windows, shown in Figure 4-3, contain a large number of tabs and fields, some of which are not used. On the other hand, the adapted UI combines the required fields from the three windows in one window, and the unnecessary fields are hidden as shown in Figure 4-4. The same approach was used to obtain adapted invoice UI. The initial invoice UI in SAP has a window for entering the invoice data and another for entering the serial numbers as shown in Figure 4-5. The adapted UI consolidates these two windows into one as shown in Figure 4-6.

Consequently, participants were asked to perform tasks using the initial and adapted UIs. Then, they were asked to answer a questionnaire describing their experience with the two

versions of the UIs. The time taken to complete each task was also recorded per participant to measure the efficiency of both UIs. The participants expressed their perceived usability of the UIs by answering the System Usability Scale (SUS) questionnaire (Brooke, 1996) and selecting the most appropriate terms from a selection of Product Reaction Cards (PRCs) (Benedek & Miner, 2002). Concerning the SUS questionnaire, it consists of ten questions that participants should answer by choosing one of the five possible answers; from strongly disagree to strongly agree. As to the PRCs, they include a set of positive and negative words from which participants in this study were asked to select a total of three descriptive terms (positive, negative, or a combination of both) to express their opinion about each UI.

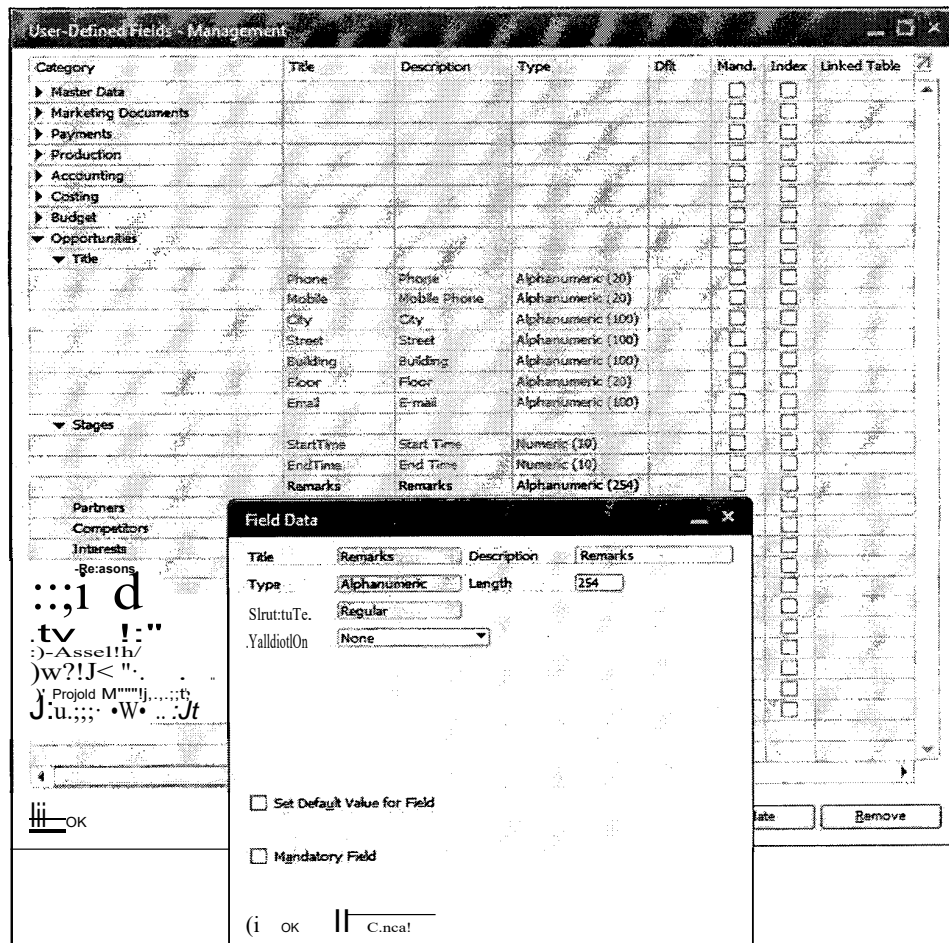


Figure 4-2 - Adding fields with User-Defined Fields Management

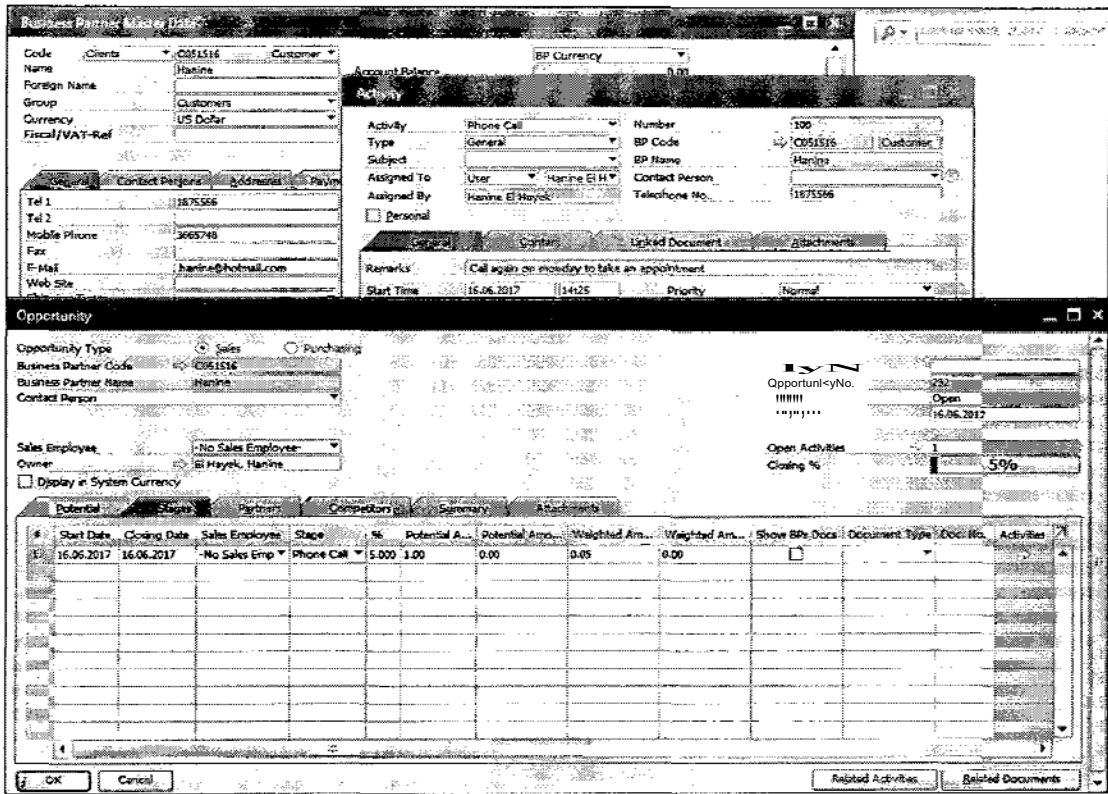


Figure 4-3 -Initial SAP sales opportunity UI

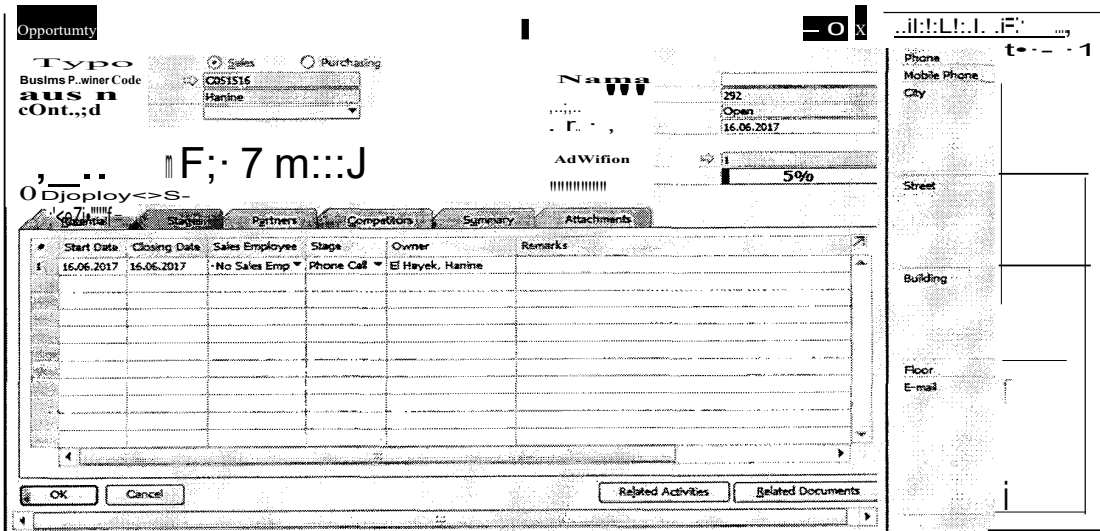


Figure 4-4 - Adapted sales opportunity UI

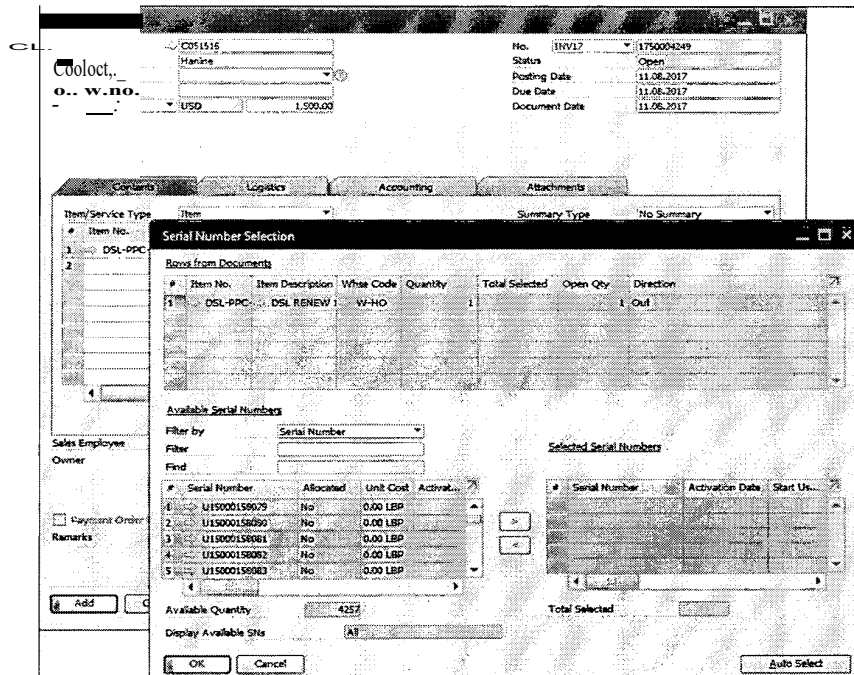


Figure 4-5 - Initial SAP invoice UI

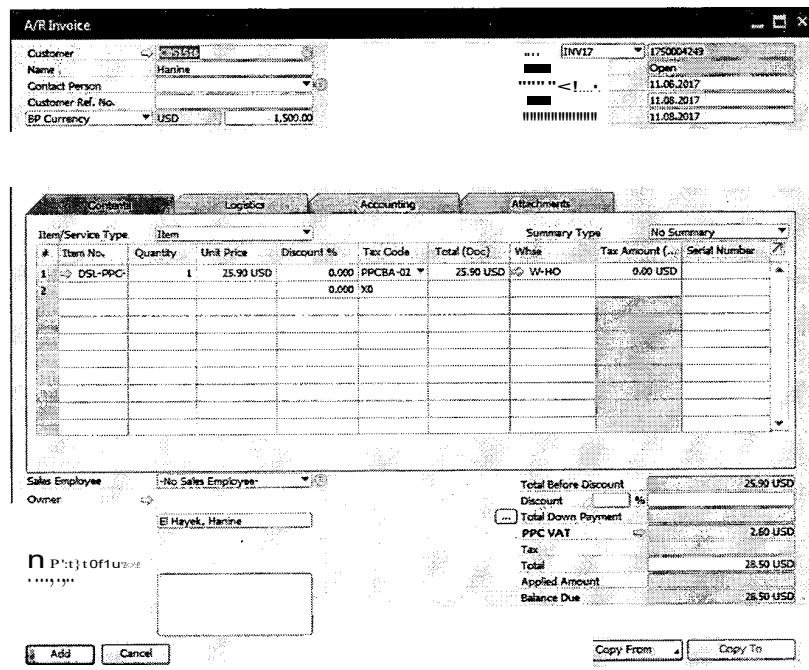


Figure 4-6 - Adapted invoice UI

## 4.2 Study Results

The results of the comparison between the initial and the adapted UIs are discussed in this section.

### 4.2.1.1 Comparison of Adapted and Initial UIs

The participants were asked to compare the adapted UIs to the initial ones by choosing one of five given response options and leaving their comments if any. These options range from 1 to 5, where 1 means that the adapted UI is much worse and 5 means that it is much better. The results of this comparison are shown in Table 4.1.

The sales participants found the adapted opportunity UI to be much better than the initial UI with an average rating of 5 which means that the ten participants selected "much better" as an answer. They said that with the adapted UI, there is no chance of overlooking fields or forgetting to enter data in a category of fields since there is one window instead of three. They said that due to having one window, everything is clearly presented in one place. Hence, the UI becomes more user-friendly and allows end users to accomplish their tasks more efficiently.

The accounting participants gave an average rating of 4.7 to the invoice UI. They restated what was said by the sales participants and added that consolidating the UIs makes work more efficient. They said that the adapted invoice UI is more practical to use than the initial SAP UI, especially when end users are required to complete a large number of transactions in a short amount of time.

**Table 4.1 – Results of comparing adapted UIs to initial SAP UIs  
(1=adapted UI is much worse to 5=adapted UI is much better)**

	Opportunity (Sales)	Invoice (Accounting)
Mean	5	4.7
Median	5	5
Standard Deviation	0	0.48
Standard Error	0	0.15

#### 4.2.1.2 Efficiency and Perceived Usability Results

The results in this section show that consolidating multiple UIs into one improves the end users' task completion efficiency.

The completion time of the different tasks are demonstrated by the box-plots in Figure 4-7 and Figure 4-8. These figures show that the time taken to finish the task with the adapted UIs is much smaller than the time needed for the initial UIs. The adapted opportunity UI reduced the task completion time to less than half as shown by the box plot in Figure 4-8. The adapted invoice UI also reduced the task completion time as shown in Figure 4-7, but the reduction was less than the one achieved by the adapted opportunity UI. The primary reason behind the difference in the reduction of the task completion time is the number of windows in the initial versions of the UIs. The initial invoice UI had two windows, while the initial opportunity UI had three windows. Hence, the adapted version of the opportunity UI showed a more significant improvement in task completion time.

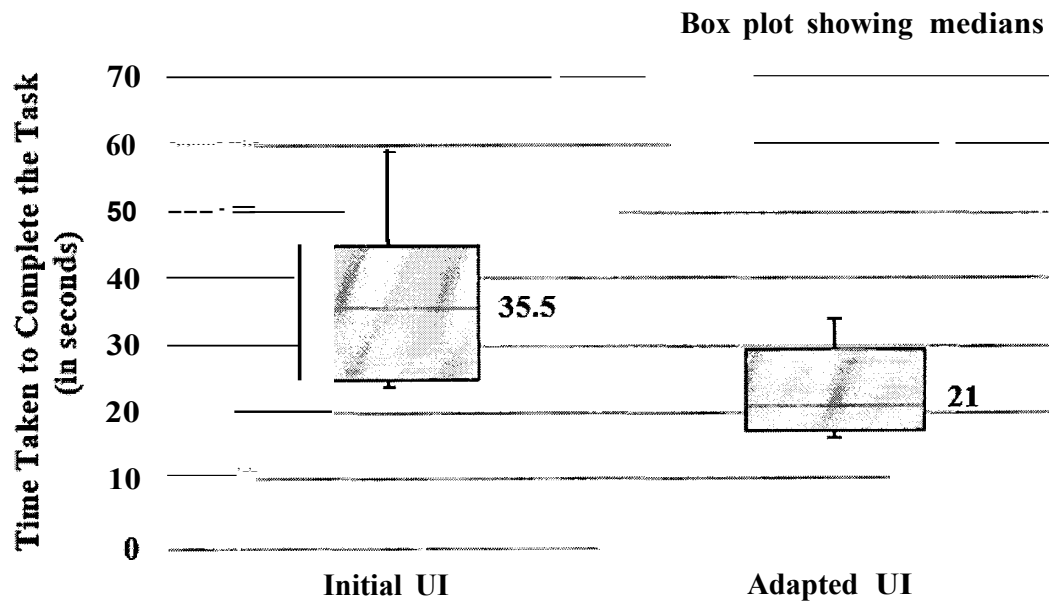


Figure 4-7 - Time taken to complete the initial and adapted invoice UIs

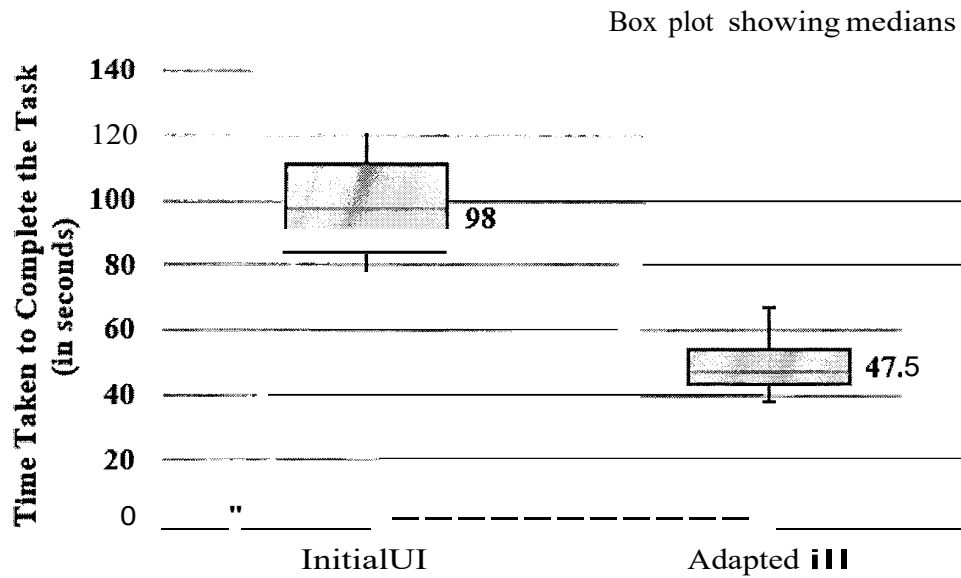


Figure 4-8 -Time taken to complete the initial and adapted opportunity UIs

The results of the participants' answers to the SUS questionnaire are illustrated by the box-plots in Figure 4-9 and Figure 4-10.

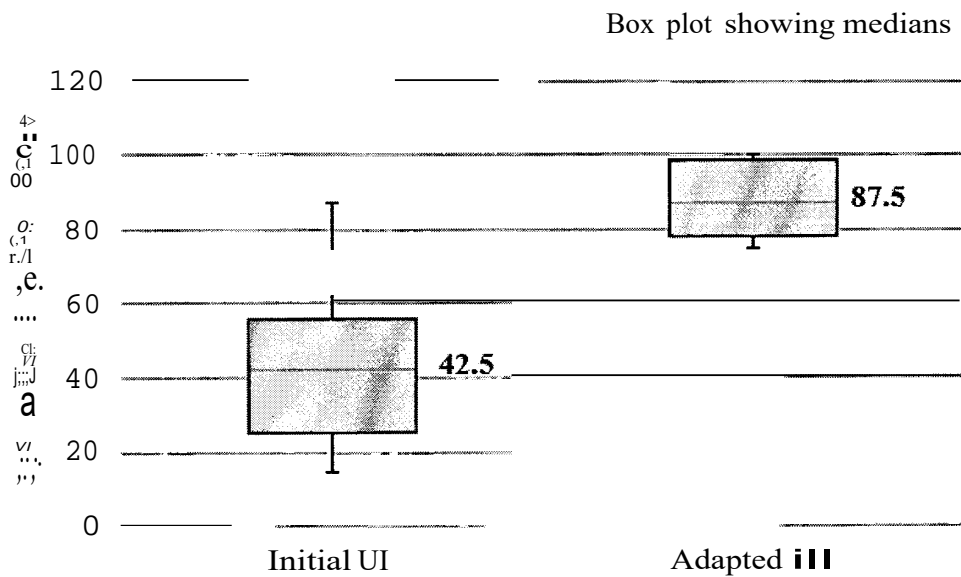
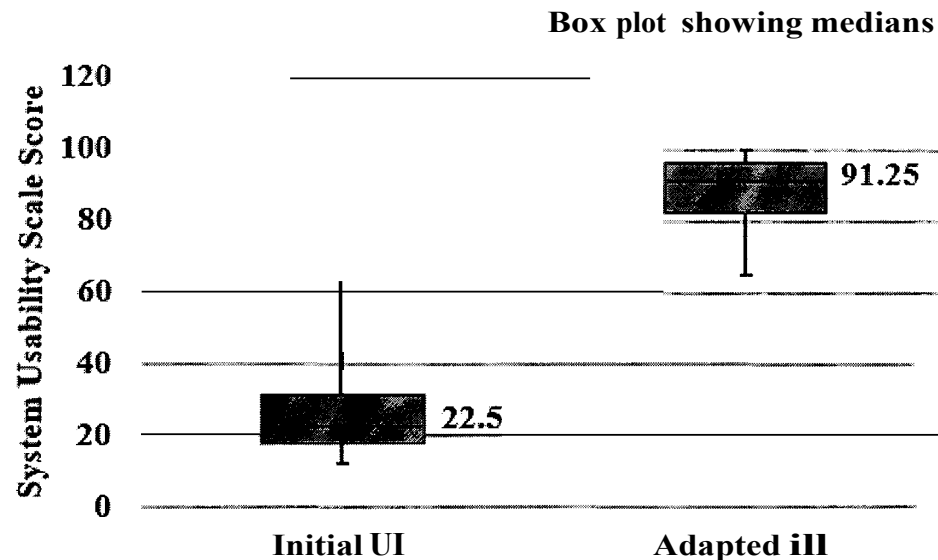


Figure 4-9 - SUS Scores for the initial and adapted invoice UIs

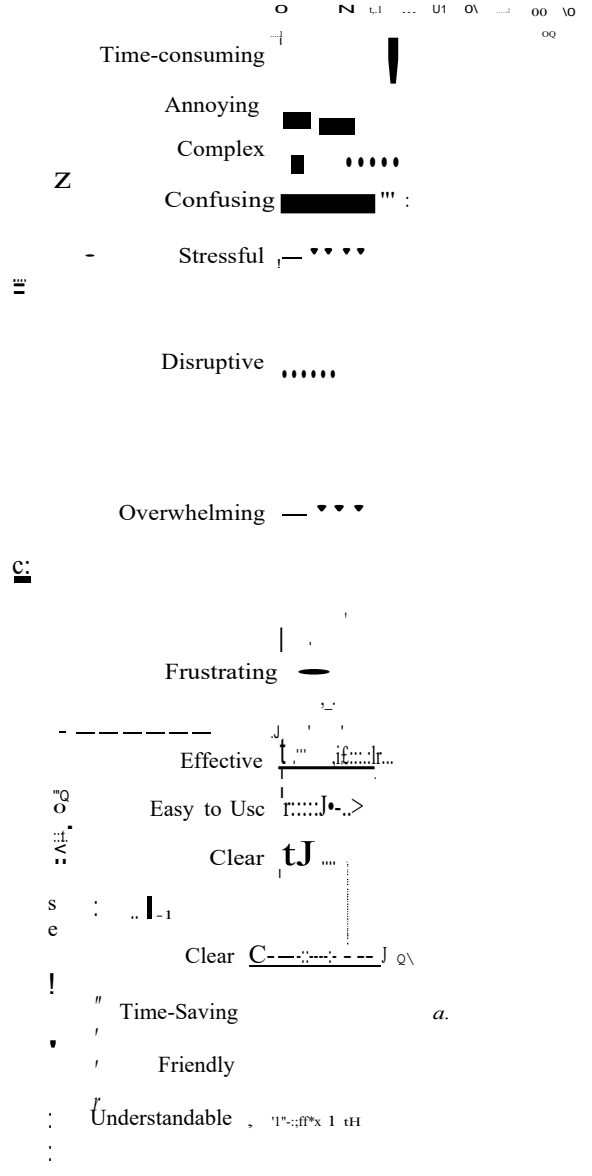


**Figure 4-10- SUS Scores for the initial and adapted opportunity UIs**

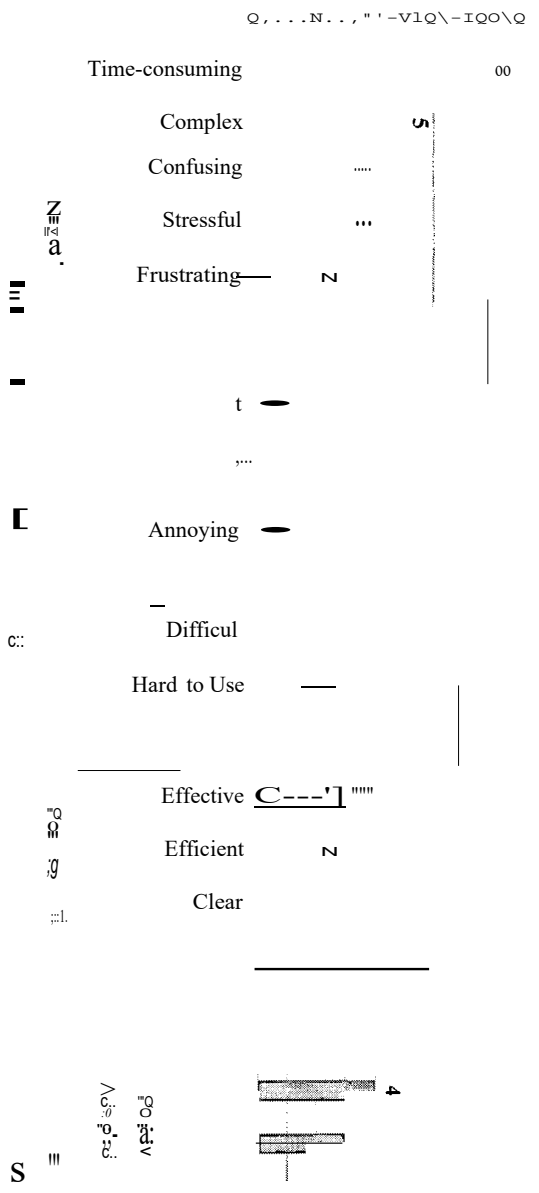
The PRCs selected by the participants are displayed for the invoice and opportunity UIs by the charts in Figure 4-11 and Figure 4-12 respectively. The terms selected for the initial UI versions were 77% negative and 23% positive for the invoice UI and 80% negative and 20% positive for the opportunity UI. On the other hand, the terms selected for the adapted versions of both UIs were 100% positive. These results show that positive and negative terms were used to describe the initial UIs; mainly a greater amount of negative terms were used. On the contrary, only positive terms were selected for the adapted UIs, which shows that there is a significant improvement in the end users' perceived usability.

To conclude, it is possible to say that the results demonstrate that the adapted UIs exhibit a significant improvement in perceived usability and efficiency in comparison to the initial versions.





Easy to Use



Easy to Use



## Chapter 5: An Approach for Developing Model-driven Multi-target Validation

This chapter presents a study that investigated error messages and types of validation operations, which are used in the SAP ERP system. The results of this study informed the design of an approach for developing model-driven multi-target UI validation.

### 5.1 Problem with Error Messages

A scan through the SAP error messages was carried out to determine the problem reported by SAP users in the usability problems presented in Section 3.2. As the users claimed, it turned out that error messages are sometimes vague, misleading, and unhelpful as shown in Table 5.1 and Table 5.2.

The samples of error messages presented in Table 5.1 are variations of a message that warns about an empty mandatory field that requires a value. However, these messages variations are indicated in ways that are unclear.

**Table 5.1 - Unclear error messages**

<b>Error message</b>	<b>Intent of message</b>
You have to define an Item Group	Missing Item Group
Blank Service Group (Dim I) / Service type (Dim2)	Missing Service Group or Service type
Accrual Cost Type should be Defined for this Item	Missing Accrual Cost Type
Invalid date. Field: Execution Date	Missing Execution Date
Missing Item Code	Missing Item Code
Payment Method is mandatory	Missing Payment Method
You have to specify the Bank	Missing Bank

The three example error messages in Table 5.2 do not specify the purpose of the message (i.e., do not indicate why is the entered data are wrong).

Table 5.2 -Unhelpful error messages

Error message	Lack of explanation
The entered DSP Fee is unacceptable	Why is the fee unacceptable?
Choose another document Series	Why should another series be chosen?
Wrong Name	Why is the name wrong?

## 5.2 Study of Error Messages and their Proposed Validation Types

A sample of 242 error messages were inspected. These messages were extracted from the following 20 Uis: Users – Setup, Posting Periods, Document Numbering Series – Setup, Business Partner, Journal Entry, Item Master Data, Exchange Rate Differences, Chart of Account, *A/R* Invoice, A/R Credit Memo, Sales Order, A/P Invoice, A/P Credit Memo, Purchase Orders, Goods Receipt PO, Goods Return, Incoming Payment, Outgoing Payment, Connection, and Payment Method. This inspection elicited the following ten types of validation rules: Exists, Comparison, Mandatory, Restrict Type and Size, Restrict Value, Regular Expression, Range, Sum, Count Distinct, Count.

The "Exists" validation rule checks the existence of the data in the system. It is used to test if the entered data already exists and if the data to be removed or modified is linked to other documents. As for the "Comparison" rule, it compares two fields or a field with a defined value and confirms if the required condition is met. The "Mandatory" rule validates that essential fields are not left blank. Regarding the "Restrict Type and Size" rule, it is a policy used to control the type and size of the inserted value in a field. The "Restrict Value" is used to restrict the input to only the allowed values defined in a list for instance. Another validation type is the "Regular Expression" which tests if the added string matches a recommended sequence of characters and patterns. The "Range" rule verifies that an input is within some defined lower and upper limits. Concerning the "Sum" rule, it is used to calculate the total sum of a field depending on required condition. The "Count" rule counts the number of records that correspond to some indicated condition. The "Count Distinct" rule works likewise but it also eliminates the repetitive appearance of the identical data.

A sample of the examined error messages with their corresponding validation types is presented in Table 5.3. The entire set of error messages from the study is presented in Appendix B. This study showed that numerous validation rules can be substituted with prevention for example the restrict value rule like the one shown in Table 5.3.

Table 5.3 -Sample error messages with their validation types

Error Message	Validation Type
This entry already exists in the following tables 'Users' (OUSR) (ODBC - 2035) [Message 131-183]	Exists
You cannot remove a business partner with a balance not equal to zero	Comparison
You have to define if this BP is a Personal Account or not	Mandatory
Mobile phone should be 8 numbers	Restrict Type and Size
The field Connection Mandatory in SIISR should beY or N	Restrict Value
The GIL Account should have the format as [NNNNN.NNNNN]	Regular Expression
You cannot Assign more than one connection per invoice	If / Count Distinct
Total Pay share should be 100%	Sum
Discount % should be between 0 and 100	Range
You cannot order more than one modem in one Application	Exists / Count

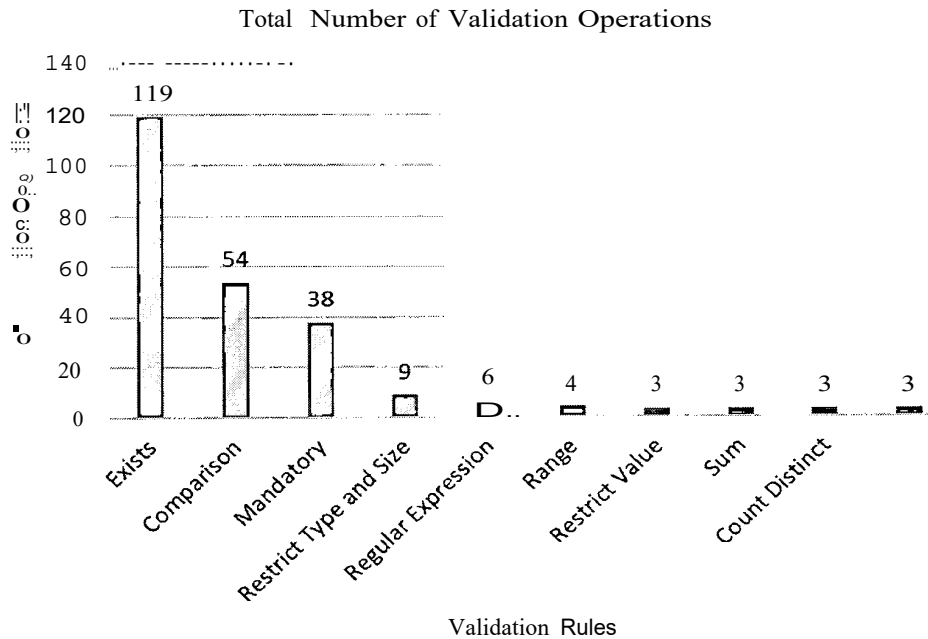


Figure 5-1 -Total number of times the validation rules are used

The frequency of usage of the validation rules and their distribution across the sampled UIs are summarized in Figure 5-1 and Figure 5-2 respectively. As shown in Figure 5-1, the highest frequency is 119 times for the "Exists" validation rule, followed by "Comparison" (54), and "Mandatory" (38). As shown in Figure 5-2, the "Sales Order", "Payment Method", and "Item Master Data" UIs use the largest number of diverse validation rules, which are 9, 6, and 5 respectively.

The boxplot of the different validation rules with their mean values is shown in Figure 5-3. The boxes display the distribution of data. The maximum number of validation operations reached is 17. The latter is for the validation rule "Exists", which has a median of 4.5 and a mean of 5.95 as designated by the statistics presented in Table 5.4. As for the "Comparison" rule, it has a median of 1 and a mean of 2.7. Nevertheless, all the other rules have a zero median. The "Mandatory" validation rule has a mean of 1.9. This mean value decreases for the remaining rules. The individual dots in Figure 5-3 are the outliers, which represent distant observation points. These points exist due to the variability in the number of operations needed in each UI.

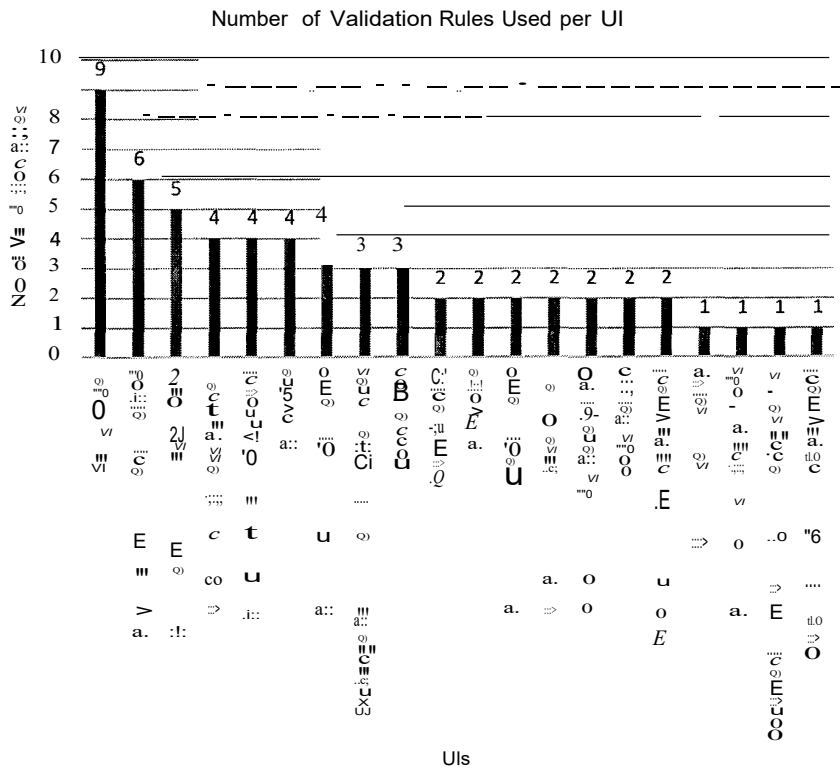
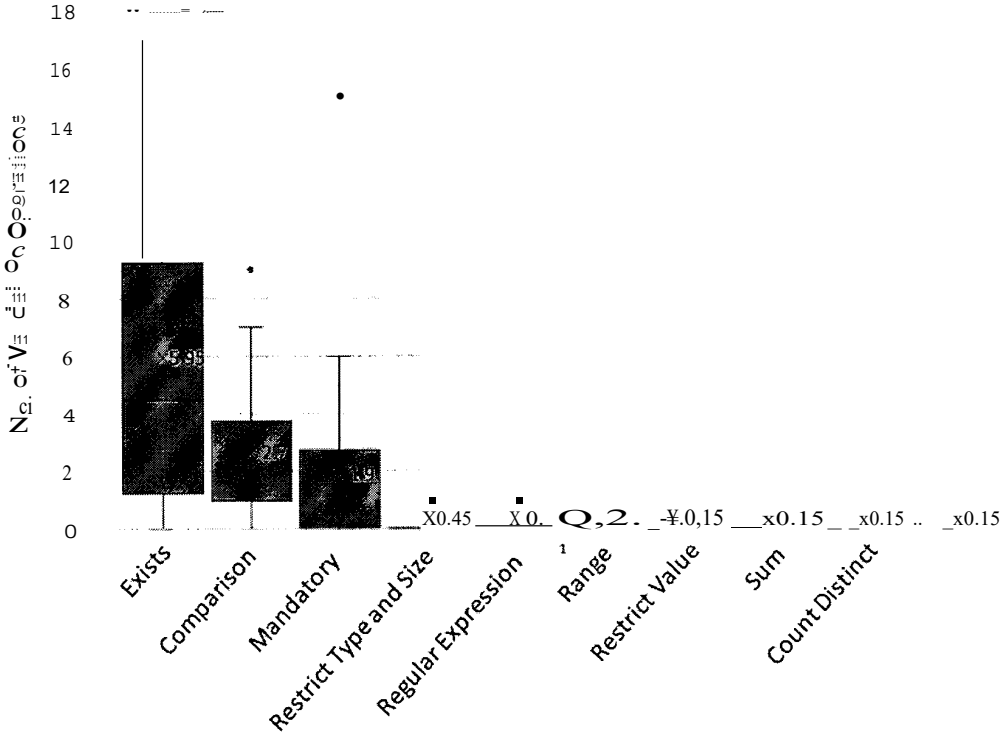


Figure 5-2 -Total number of different validation rules used per UI

**Table 5.4 - Descriptive statistics of the validation rules**

	Mean	Median	Standard Deviation
Exists	5.95	4.5	5.42
Comparison	2.7	1	3.29
Mandatory	1.9	0	3.57
Restrict Type and Size	0.45	0	1.23
Restrict Value	0.15	0	0.49
Regular Expression	0.3	0	1.13
Range	0.2	0	0.7
Sum	0.15	0	0.67
Count Distinct	0.15	0	0.37
Count	0.15	0	0.49

Box plot with mean values



**Figure 5-3 - Number of validation operations per rule**

### 5.3 Proposed Validation Model Concepts

The high-level concept diagram of the proposed validation approach is presented in Figure 5-4. It consists of a *Container* class having a *Title* attribute which represents the validation rule name. A *Container* can contain zero-to-many *Labels* that display a *Value*. It can also contain zero-to-many *Inputs* that have a *Value* and *SuggestedValue*. Additionally, zero-to-many *Messages* can be held within a *Container*. These *Messages* are multilingual (i.e. support multiple human languages). They consist of friendly *UserMessages* for end-users as well *ErrorCodes* and *TechnicalMessages* for developers. Furthermore, a *Container* can have zero-to-many *LogicalOperators* (And/Or), and can be preceded or succeeded by a maximum of one *LogicalOperator*. A *Container* can also contain zero-to-many children that are also *Containers*. This is used to compose nested rules.

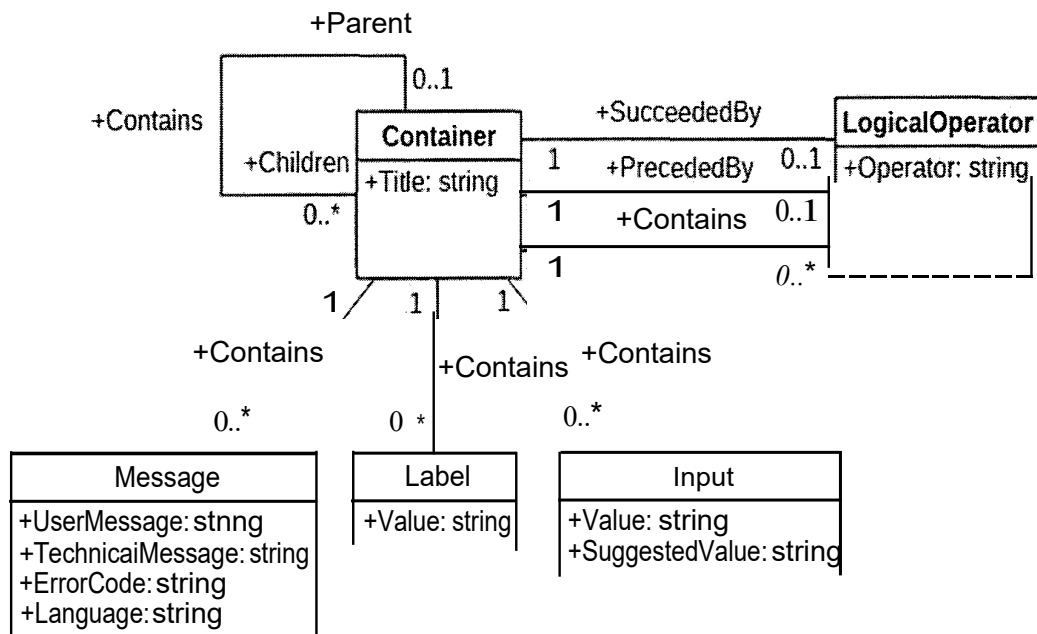


Figure 5-4- High-level concept class diagram

The proposed validation rules are presented by the class diagram shown in Figure 5-5. Each *ValidationRuleSet* is composed of at least one *ValidationRuleGroup*, which is in turn formed from at least one *ValidationRule*. The latter can be one of the following: *ComparisonRule*,



*MandatoryRule*, *RangeRule*, *RegularExpressionRule*, *RestrictTypeAndSizeRule*, *RestrictValueRule*, and *QueryValidationRule*. They all inherit the attributes and method of the parent class *ValidationRule*. Besides, the *ExistsRule*, *CountRule*, *CountDistinctRule*, and *SumRule* are subclasses of the *QueryValidationRule*. The *Condition* is part of the superclass *QueryValidationRule*. A *Condition* can have zero-to-many *Comparison*, *Range*, and *RegularExpression* rules. Furthermore, the *ValidationRuleGroup* can contain zero to many programming constructs that include "IF" conditions.

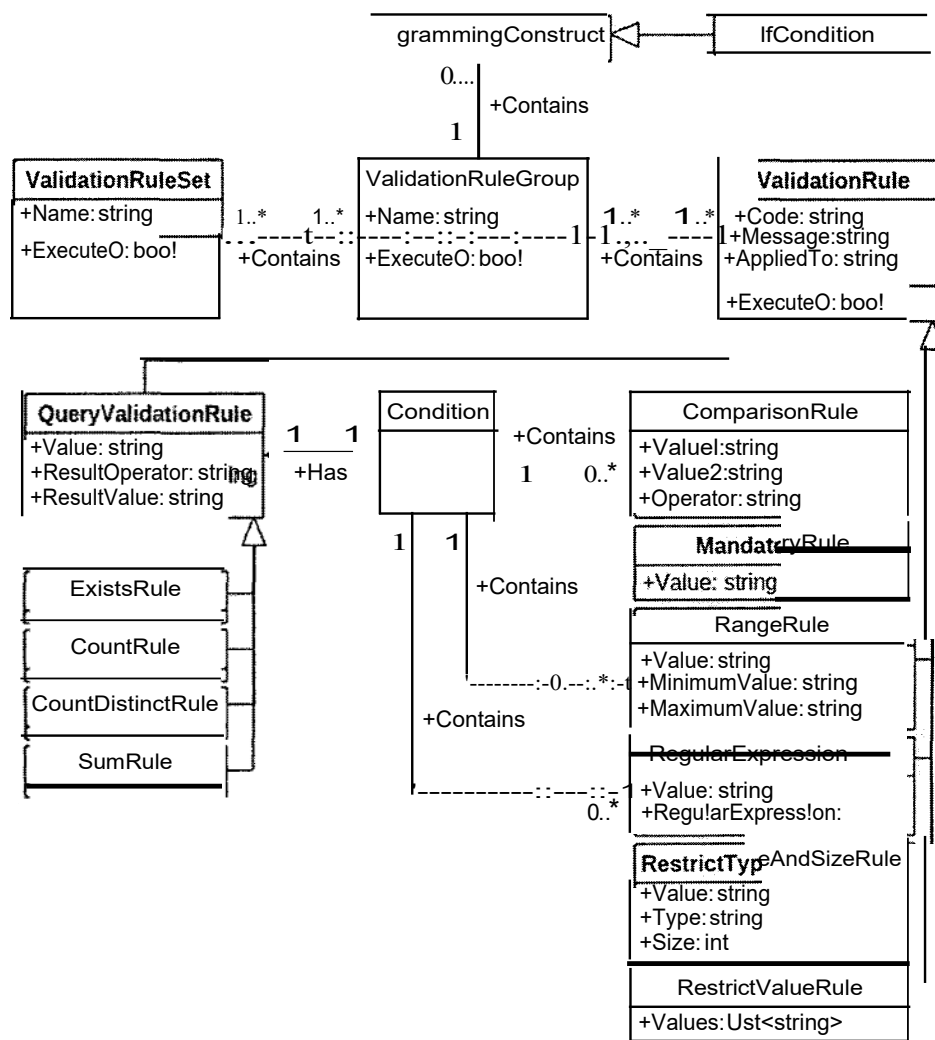


Figure 5-5 - Validation rules class diagram

## 5.4 Multi-target Validation

Validation can be executed on the following levels: UI, domain model, and database. In the proposed approach, developers can choose on which level(s) the validation should be done. For example, it can be performed only on the domain level because it is the basis of the database and the UI. Nonetheless, validation can also be performed in the UI to provide direct validation while the user is working. Validation can also be performed on the database to prevent undesirable entries that are submitted from a variety of software applications. A sales order activity is taken as an example to present the validation on the three different levels.

### 5.4.1 Domain Model

A sales order is an order confirmation document issued by the company to a customer. The domain model of the sales order is illustrated in the class diagram shown in Figure 5-6. This model includes the *SalesOrder* that contains the following information: OrderNumber, PostingDate, DueDate, CurrencyCode, TradeDiscount, Subtotal, VatAmount, Total, CustomerCode, CustomerName, Email, DSLNumber, and ConnectionStatus. The enumeration *ConnectionStatus* defines the set of status values that are used in the following examples. A *SalesOrder* has exactly one *Address*. This *Address* consists of a Country, County, City, Street, Building, Floor, and PostCode. The *SalesOrder* is made up of at least one *SalesOrderLine*. Each *SalesOrderLine* in the *SalesOrder* has a unique Lineld and can contain only one *Item* along with the Quantity, UnitPrice, VatAmount, MarkDown, and ExtendedPrice. The *Item* has an ItemId, ItemCode, Name and UnitPrice.

### 5.4.2 Database

At the database level, Figure 5-7 shows the relational model that graphically demonstrates the database structure of the sales order example. This diagram is mapped to the domain model (refer to Section 5.4.1).

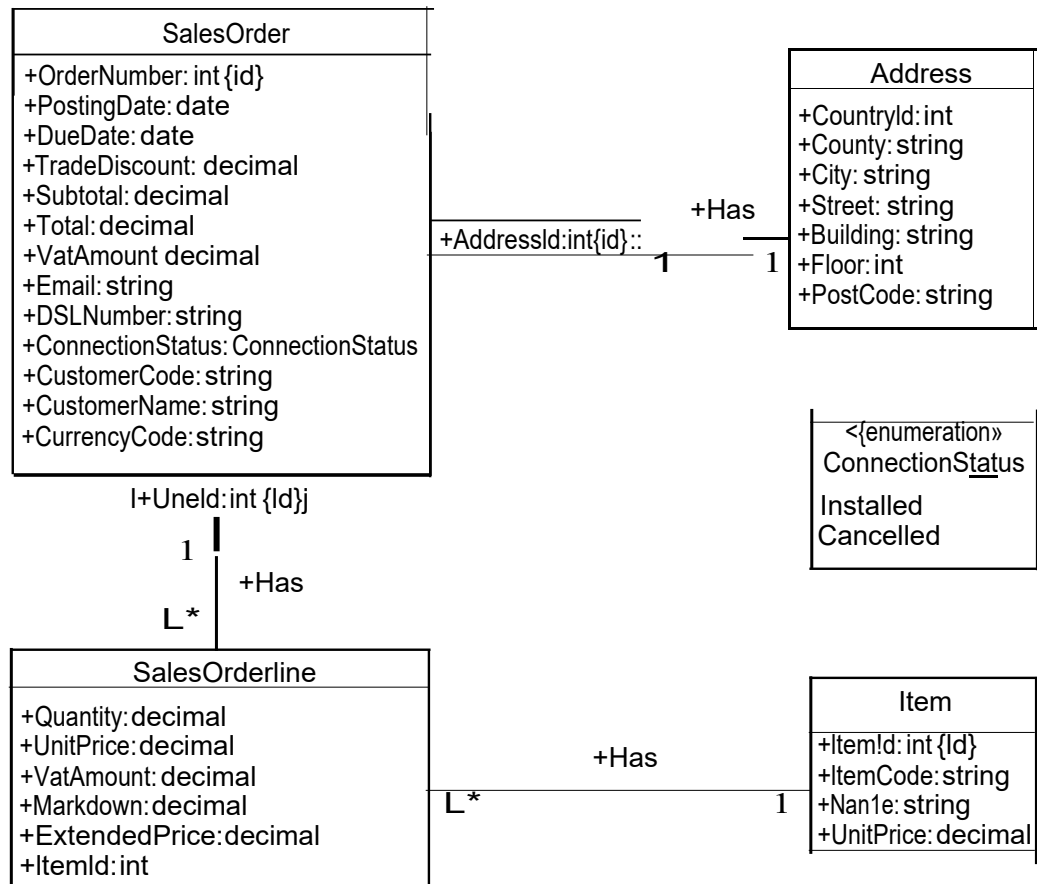


Figure 5-6 - Domain model of the sales order

### 5.4.3 User Interface

The Cameleon Reference Framework (Calvary et al., 2003) is a unified framework, which suggests the representation of UIs on the following levels of abstraction:

- a) The task model is the highest level of abstraction and is platform-independent. It is represented using ConcurTaskTrees (CTT) notation (Paterno et al., 1997). This level shows the UI characteristics as tasks done by the system and the users. These tasks are linked using operators that express ordering restrictions and prerequisites.
- b) The abstract UI model (AUI) is the portrayal of a user interface that is free from any kind of modality like graphical and voice.

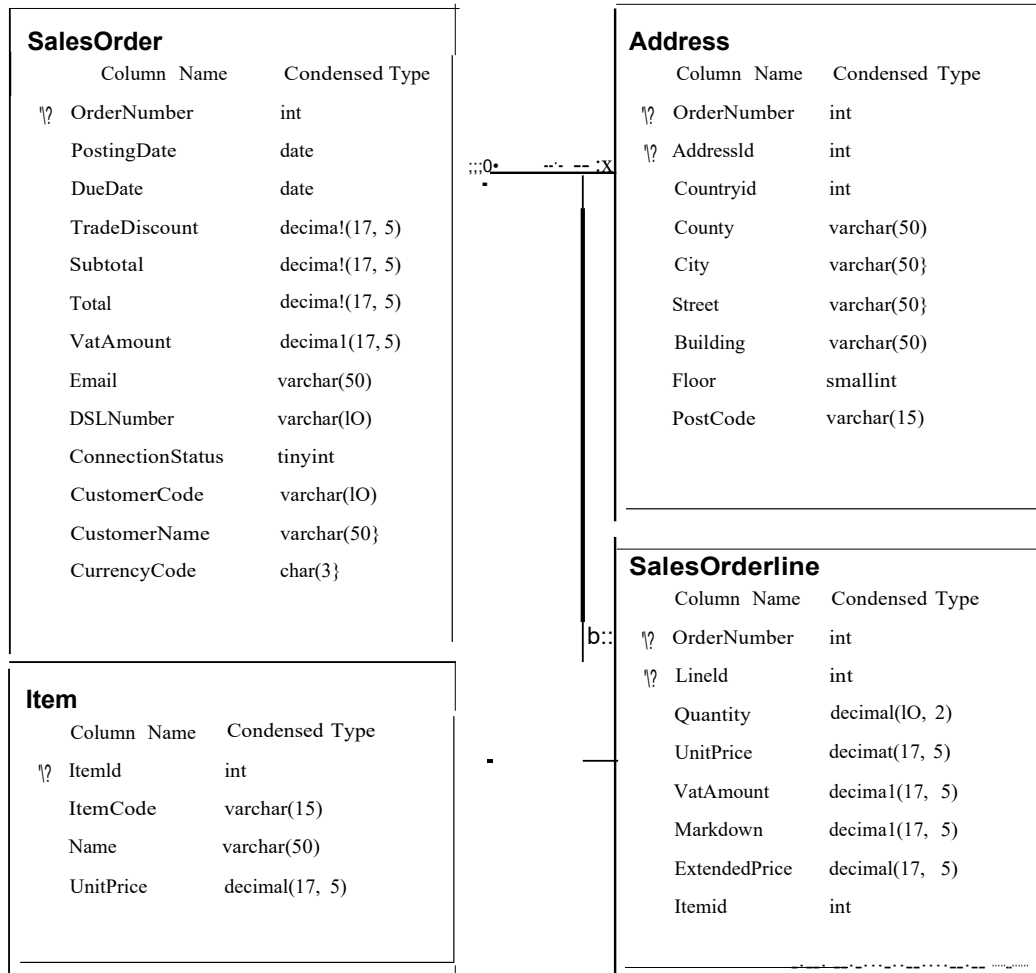


Figure 5-7 - Database diagram of the sales order

- c) The **concrete UI model (CUI)** is the presentation of a modality-dependent UI. For instance, this level may contain widgets such as: text boxes, buttons, labels, radio buttons, check boxes, and so on.
- d) The **final UI model (FUI)** is the actual UI presentation resulting from code generation or runtime interpretation. The FUI is presented using a presentation technology such as HTML.

The example shown in Figure 5-8 represents an excerpt of a sales order UI presented on the levels of abstraction proposed by the Cameleon Reference Framework.

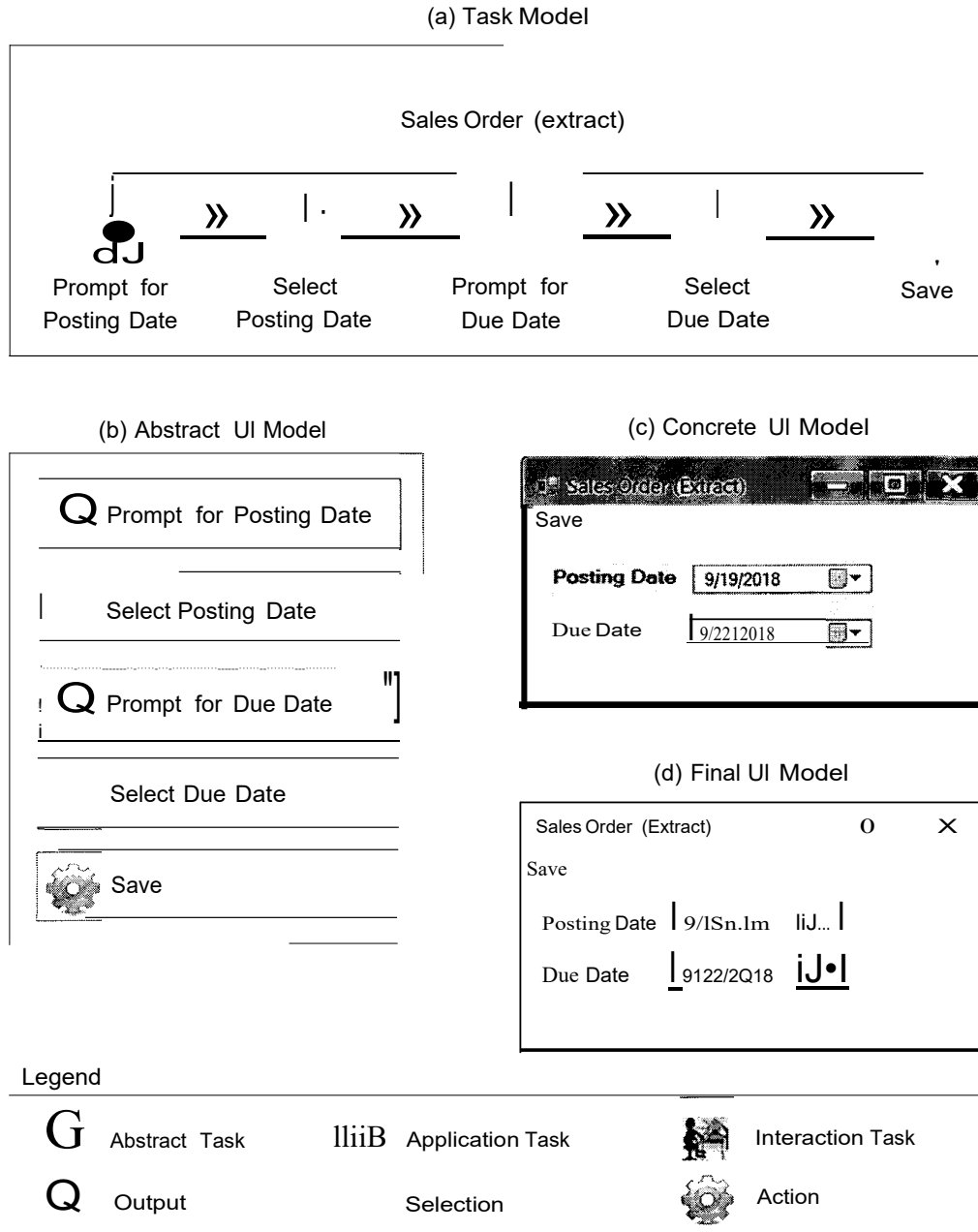


Figure 5-8- Example of sales order (extract) UI represented on multiple levels of abstraction: (a) task model, (b) AUI, (c) CUI, and (d) FUI

### 5.4.4 Validation Model

The comparison validation model for the sales order example is represented by the rule shown in Figure 5-9. This rule compares two fields depending on the selected condition. The two fields can be read from the domain model (Section 5.4.1), the database (Section 5.4.2), or the UI (Section 5.4.3).

In this example, the model checks if the posting date of a sales order is less than or equal to its due date. If the answer is true, then the sales order can be saved. Otherwise, an error message is displayed with the suitable language for the end user to understand. This message, which is in this example "The Posting Date cannot be greater than the Due Date", informs the user of the wrong date entries that should be adjusted. Once the dates are corrected, the user can save the document.

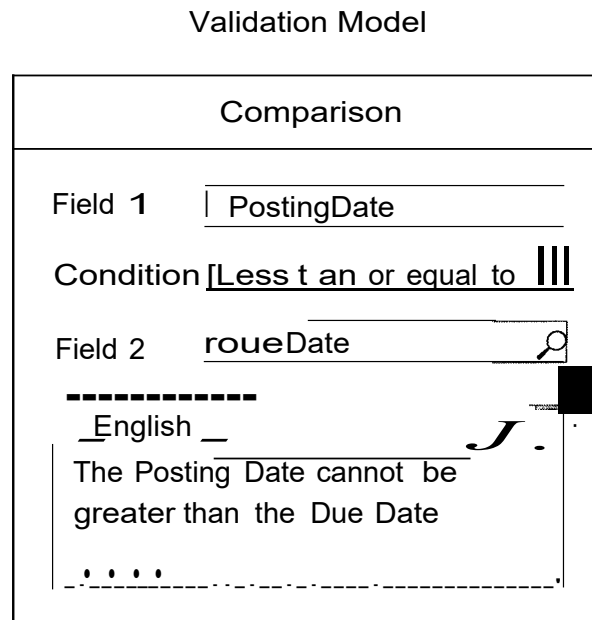


Figure 5-9 -The comparison rule for the sales order extract example

The validation model can be triggered when there is an input, selection, or action. The option can be set in the setup of the system to specify whether the model is provoked only on the activation of an action or on the addition, selection, and update of a value.

In the sales order extract example, shown in Figure 5-10, the comparison rule is triggered when there is a change in the posting date and due date and when the save button is pressed.

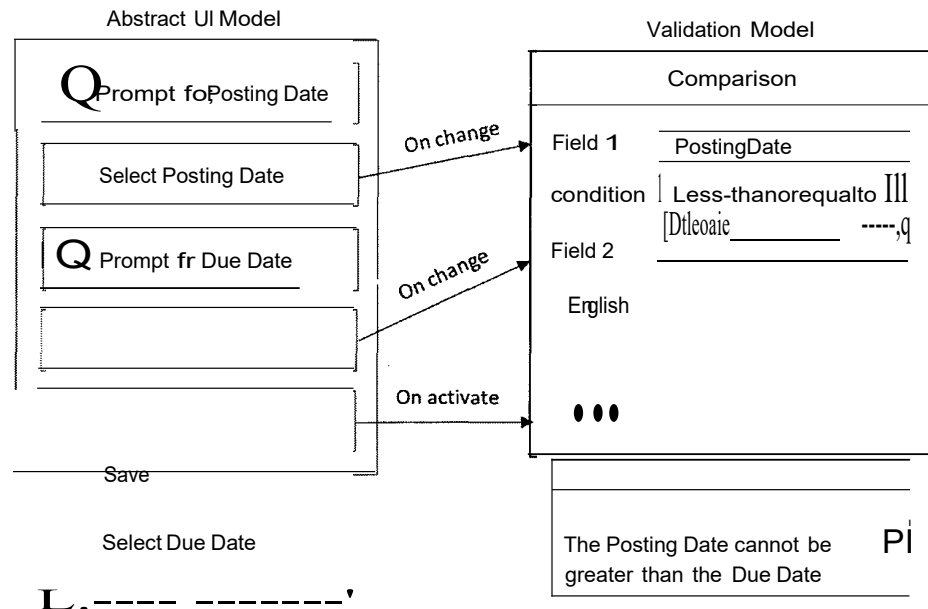


Figure 5-10 -Triggering the validation model on change and activation

The validation model can be used to generate executable validation that operates on the domain model (classes), on the database, and on the user interface as shown by the examples presented in Sections 5.4.4.1, 5.4.4.2, and 5.4.4.3 respectively. These examples are based on the validation model presented in Figure 5-10. Here it is important to note that these examples represent what could potentially be generated, assuming that code generators are implemented to complete the approach presented in this thesis.

#### 5.4.4.1 Example of validation on the domain

The C# code example in Listing 5.1 uses fluent validation to validate "SalesOrder" class.

Listing 5.1 - Example of validation on a domain model class

```
public class SalesOrderValidator : AbstractValidator<SalesOrder>
{
    public SalesOrderValidator()
    {
        RuleFor(so => so.PostingDate)
            .LessThan(so => so.DueDate)
            .WithMessage("The 'Posting Date' cannot be greater than
                the 'Due Date'.");
    }
}
```

#### 5.4.4.2 Example of validation on the database

The SQL code example in Listing 5.2 creates a trigger that validates records that are being inserted into the "SalesOrder" table. An exception is thrown in case the posting date is greater than the due date.

**Listing 5.2 - Example of validation on a database**

```
Create Trigger [dbo].[Trigger_Insert_SalesOrder]
On [dbo].SalesOrder
For Insert
AS
Begin
    if(exists(Select * From inserted Where PostingDate > DueDate))
        begin
            Throw 50001,'The ''Posting Date'' cannot be greater
            than the ''Due Date''.', 1
            Rollback Transaction
        end
End
```

#### 5.4.4.3 Example of validation on the UI

The UI validation example in Listing 5.3 is written in C# and operates on the Windows Forms presentation technology. Event handlers are used to detect changes in the values of the posting and due dates, then a validation function is called to show an error if needed.

**Listing 5.3 - Example of validation on the UI**

```
private void dtPostingDate_ValueChanged(object sender, EventArgs e)
{
    ValidatePostingDate(dtPostingDate);
}

private void dtDueDate_ValueChanged(object sender, EventArgs e)
{
    ValidatePostingDate(dtDueDate);
}

private void ValidatePostingDate(DateTimePicker dt)
{
    if (dtPostingDate.Value > dtDueDate.Value)
    {
        errorProvider.SetError(dt, "The 'Posting Date' cannot be
        greater than the 'Due Date'.");
    }
}
```



## 5.5 Examples of Validation Rules

This section includes some examples of the suggested validation rules that apply to the sales order example presented in Section 5.4.

Figure 5-11 shows the "mandatory" rule that determines the missing required information for a document. For instance, to add a sales order, it is essential to put the customer's name hence the message "Missing Customer Name" is shown if this field is empty.

The diagram is enclosed in a rectangular border. At the top, the word "Mandatory" is centered in a bold font. Below this, the text "Field CustomerName" is displayed, with "CustomerName" underlined. Underneath, there is a dropdown menu containing the text "E lish" and a small square icon with a checkmark. Below the dropdown, the text "Missing Customer Name" is shown, followed by a magnifying glass icon and three dots " . . . " at the bottom.

Figure 5-11 - Mandatory rule example

On the other hand, the "restrict type and size" rule specifies the type and the number of characters a field can hold. For example, the DSL number is the same as the standard landline number in the customer's house or office. This landline number should be an integer made up of 8 characters as it was set in Figure 5-12. In case this condition is not applicable, the user gets an error message and is forbidden to save the order until the requirements are met.

Restrict Type and Size	
Field	DSLNumber
Type	integer
Length	8
English DSLNumber should be an integer of length 8	

Figure 5-12- Restrict type and size rule example

The "restrict value" rule examines if the inputted value is among the allowed listed values like the defined document currency values shown in Figure 5-13. When the entered CurrencyCode is different from the available valid options, the error message is displayed.

Restrict Value	
Field	..... .21
Value	LBP
English Wrong value- value is not from the list	

Figure 5-13 - Restrict value rule example

A "regular expression" rule verifies certain characters or patterns in a string. An email for instance is checked using a regular expression. The "Email regex" shown in Figure 5-14 is selected by non-developer users, and it refers to the developer's hidden code for an email

regular expression. In case the inserted email is inappropriate, the message "Invalid email address" is shown.

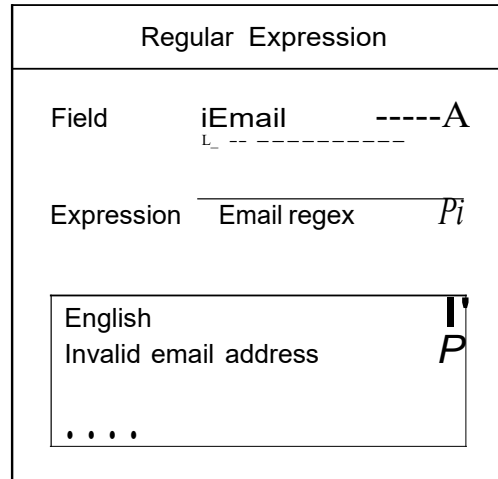


Figure 5-14- Regular Expression rule example

A "range" outlines the permitted zone of variation on a certain scale. For example, Figure 5-15 shows the range rule that sets the boundaries of the trade discount percentage. The discount ranges between zero and a hundred. If the assigned percentage is negative or greater than hundred, the user gets an error message explaining the fault entry.

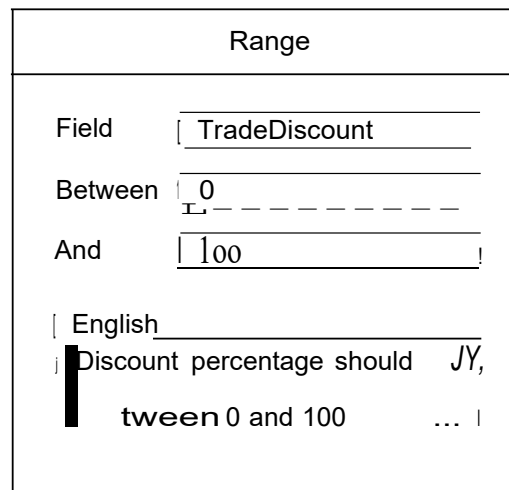
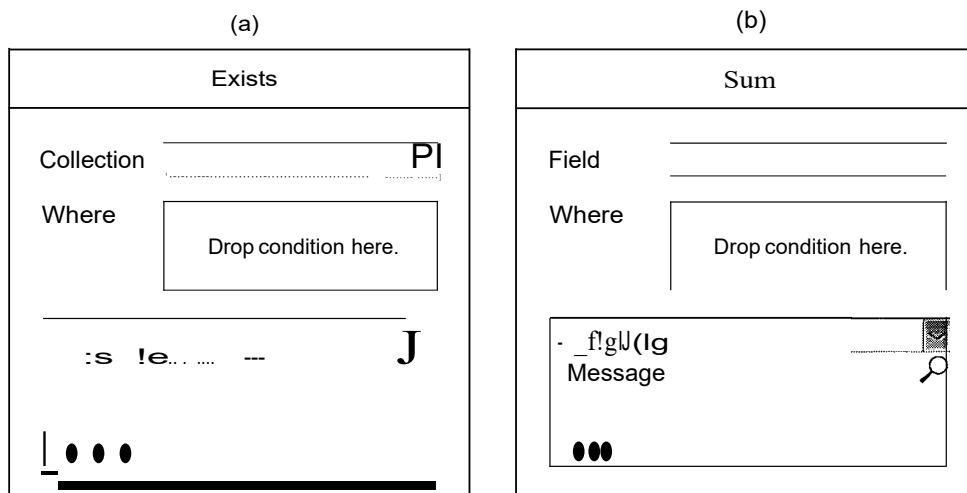


Figure 5-15 - Range rule example

Query validation rules are more complex. Some can take a collection as an input. The latter can be more than one field, e.g., a table, a list, or a UI. Additionally, the required condition can be made up of one or more validation rules connected with logic operators. The general validation models of the query validation rules "Exists" and "Sum" are illustrated in Figure 5-16. The models of the "Count" and "Count Distinct" rules are similar to the model of the "Exists" rule.



**Figure 5-16-General models of query validation rules: (a) Exists rule (b) Sum rule**

The example presented in Figure 5-17 shows the "Exists" rule that is applicable to the sales order. If the DSL number exists in a previously saved sales order with a connection status "not cancelled", the user is informed to check the problem. This includes validating the DSL number with the customer and changing it because only a single sales order can be submitted per phone number. Thus, going into more details, the "Exists" rule, in this case, contains two comparison rules linked with an "AND" operator in its condition part. The first comparison rule checks if the same DSL number is found in another sales order. The second rule checks if the connection status of that sales order is cancelled. If the two conditions are true, the sales order can be added otherwise an error is shown.

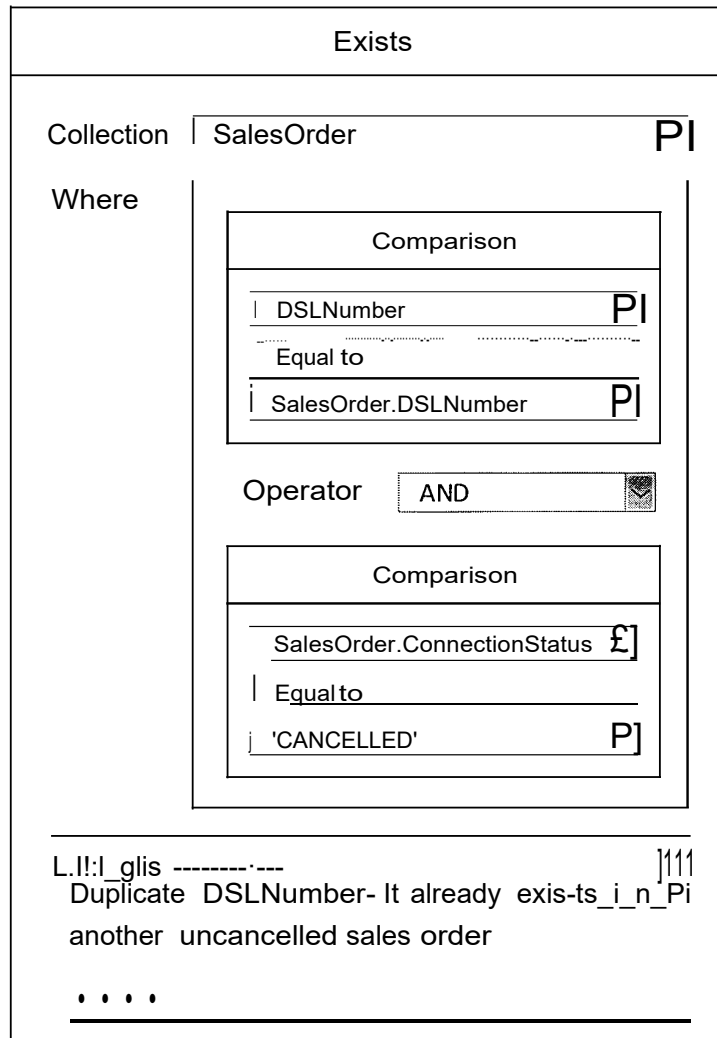


Figure 5-17- Exists rule example

The "Count" rule example shown in Figure 5-18 counts the number of sales order lines that are not empty and thereby checks if the sales order contains an item. The user is required to add at least one line to a sales order before saving it. A "Count" of zero indicates that no lines were added to the order, so the error message is displayed.

The **"If"** statement is used to check if a certain condition is met. If the condition is true, then the process can continue. On the other hand, if it returns false, the user is prompted to solve the issue. The example shown in Figure 5-19 is used to check if there is a duplicate

item in the sales order. The "Count Distinct" produces the number of different items used in the sales order lines. This number is compared to the total number of the sales order lines resulting from the "Count" that was discussed earlier. If the total number of distinct items is equal to the total number of sales order lines, then the sales order can be created. Otherwise, a message is issued to indicate that there are duplicate items.

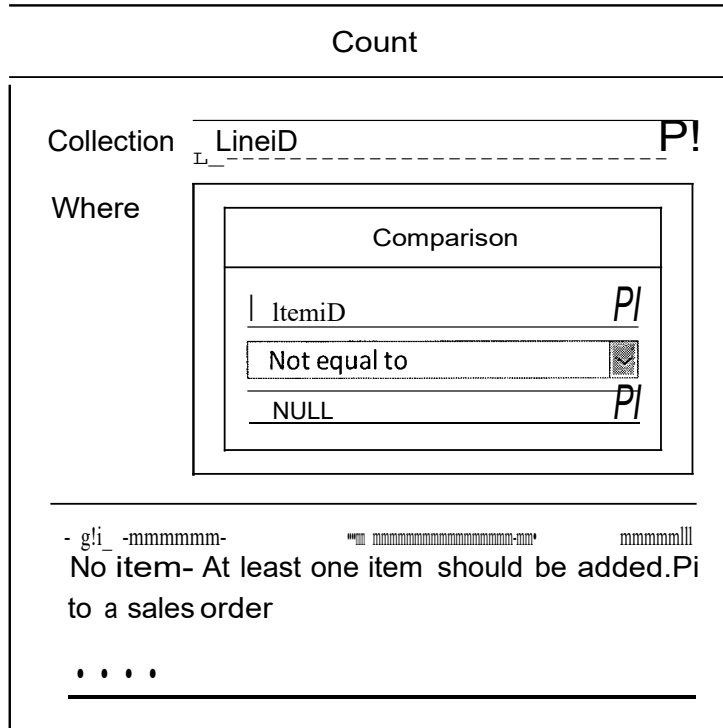


Figure 5-18 - Count rule example

As for the example shown in Figure 5-20, it checks if the sum of the extended price values from the sales order lines is equal to the total of the sales order when no trade discount is applied. The "Sum" rule is used to add all the extended price values when there is an item and no trade discount is set for the sales order. The "If" statement compares the outcome of the "Sum" with the sales order's total. If these two values are equal, then the calculation is correct and user can proceed with the sales order. Otherwise, the user is prompted to check and fix the numbers.

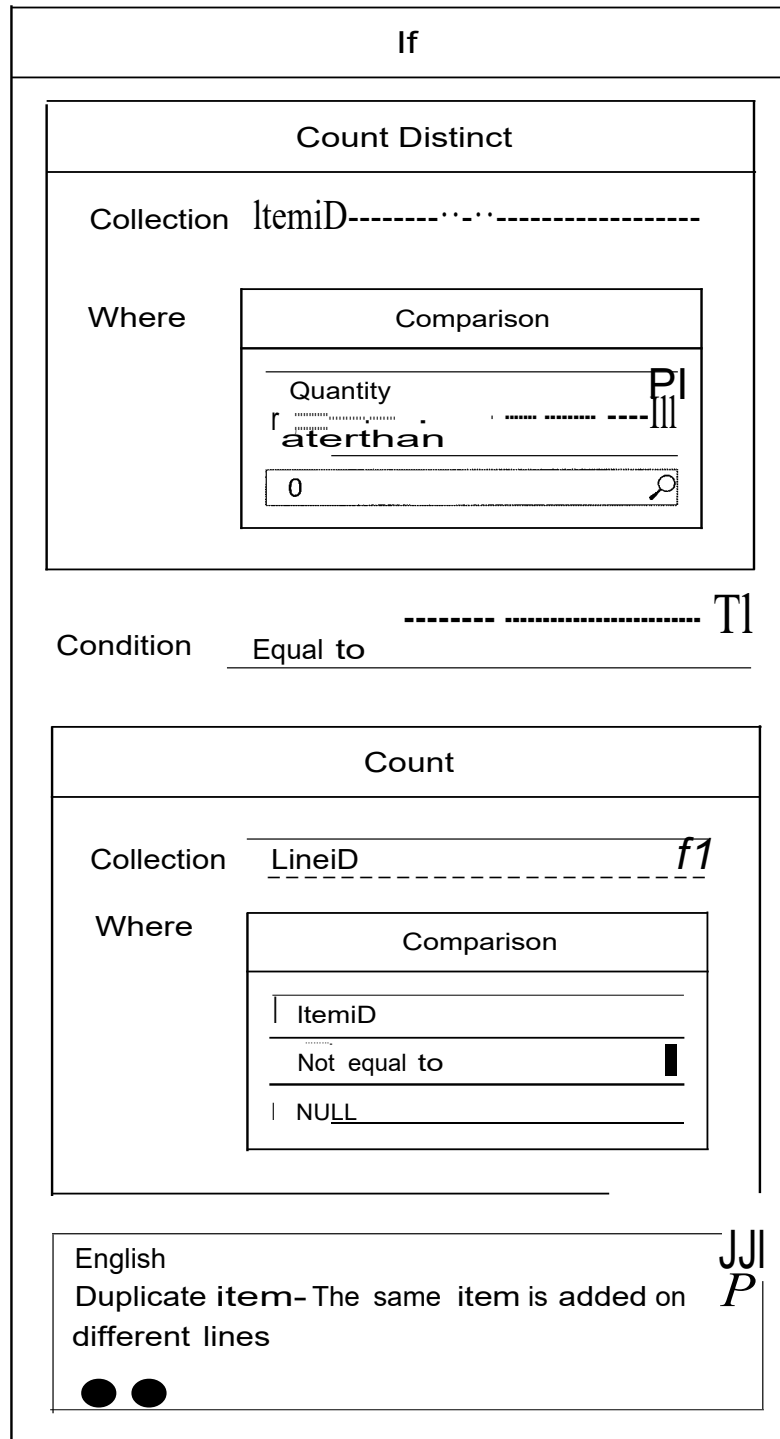


Figure 5-19- Count Distinct rule example

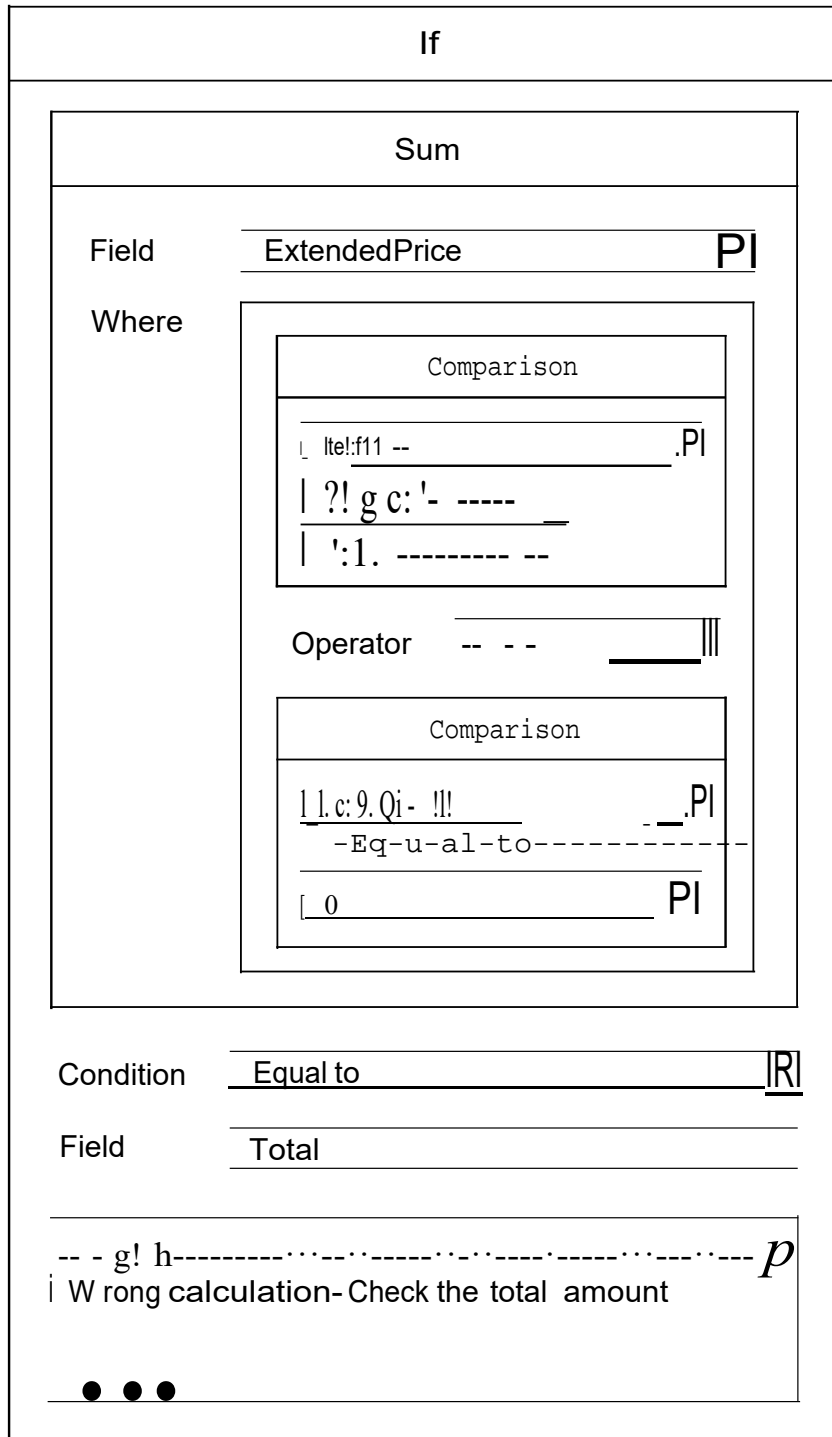


Figure 5-20 - Sum rule example



## Chapter 6: Conclusion

This chapter summarizes the main results of the studies and the core contributions of the thesis. Moreover, it proposes possible future work.

### 6.1 Main Contributions

In this thesis, a study was performed to check ERP usability issues that impact end users' efficiency and satisfaction. Two common problems were revealed as explained in Chapter 3. The first problem is the multiple UIs navigation issue and the second is the unclear error codes and messages issue.

Regarding the first problem, a study was conducted (Chapter 4) to evaluate the effect that merging UIs has on usability. The difference between the initial and adapted UIs was compared. After working with both UI versions, the participants rated the adapted UIs to be much better than the initial ones. They found the adapted UIs to be clearer, time-saving, and easy to use. They also noted that the adapted UIs do not place a heavy burden on a user's memory. In addition, the time to accomplish a task was measured for both initial and adapted UIs. It was shown that the time needed to complete a task with the adapted UIs is much less than that needed when working with the initial ones. Thus, with the adapted UIs, there is an enhancement in perceived usability and efficiency.

In Chapter 5, error messages from multiple SAP Business One UIs were analyzed to check the second problem related to unclear errors. Some errors proved to be misleading, unhelpful, and not understandable. Upon this analysis, ten distinct validation types were identified. These validation types were used in the proposition of an approach for composing validation rules. This approach relies on a visual paradigm for rule composition, which can be used by developers and non-developers. This validation approach aims to minimize the technical knowledge required to compose validation rules and to support validation on different levels including UIs (presentation layer), software classes (domain model), and databases.

## 6.2 Possible Extensions and Future Work

There are plenty of modifications that could be done to ERP systems to obtain better usability and facilitate users' work. One of these modifications, as indicated by the multi-UI study, is to make ERPs more flexible. That is, UIs must contain all the necessary information in order to reduce the number of UIs needed to accomplish a task. Adding to that, UIs should be designed in a way so that they can be adapted easily based on users' needs.

More work can be done on the multi-target validation approach that was proposed in this thesis. An engine can be created to generate and execute the validation rules. This engine would be supported by a tool that allows the definition and modification of validation rules. Furthermore, a study can be conducted in order to test the validation rules with a real-life application. Although this thesis primarily targets ERP systems, the validation approach presented here is extensible for other types of applications. Hence, different types of applications can be used in the evaluation of future extensions of this approach.

## Bibliography

- Adenle, C. (2011, July 26). 12 Reasons Why Employees Resist Change in the Workplace. Retrieved January 22, 2018, from <http://catherinescareercomer.com/2011/07/26/12-reasons-why-employees-resist-change-in-the-workplace/>
- Ahmed, M.A. (2017). The Importance of the Organizational Structuring and Departmentalization in Workplace. *The Journal of Middle East and North Africa Sciences*, 3(3), 30-38.
- Babaian, T., Lucas, W., Chircu, A., & Power, N. (2016). Evaluating Interactive Visualizations for Supporting Navigation and Exploration in Enterprise Systems: (pp. 368-377). SCITEPRESS- Science and Technology Publications. <https://doi.org/10.5220/0005770203680377>
- Babaian, T., Lucas, W. T., Xu, J., & Topi, H. (2010). Usability through System-User Collaboration. In *Proceedings of the 5th international conference on Global Perspectives on Design Science Research DESRIST* (pp. 394-409). Berlin, Heidelberg: Springer.
- Benedek, J., & Miner, T. (2002). Measuring Desirability: New methods for evaluating desirability in a usability lab setting. *Proceedings of Usability Professionals Association, 2003*(8-12), 57.
- Bingi, P., Sharma, M. K., & Godla, J. K. (1999). Critical Issues Affecting an ERP Implementation. *Information Systems Management*, 16(3), 7-14.
- Brooke, J. (1996). SUS-A quick and dirty usability scale. *Usability Evaluation in Industry*, 189(194), 4-7.
- Buckhout, S., Frey, E., & Nemeč Jr, J. (1999). Making ERP Succeed: Turning Fear into Promise. *IEEE Engineering Management Review*, 19, 116--123.
- Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Bouillon, L., & Vanderdonck, J. (2003). A Unifying Reference Framework for multi-target user interfaces. *Interacting with Computers*, 15(3), 289-308. [https://doi.org/10.1016/S0953-5438\(03\)00010-9](https://doi.org/10.1016/S0953-5438(03)00010-9)
- Escott, E., Strooper, P., King, P., & Hayes, I. J. (2012). Model-Driven Web Form Validation with UML and OCL. In A. Harth & N. Koch (Eds.), *Current Trends in Web Engineering* (pp. 223-235). Springer Berlin Heidelberg.
- Florin, M., & Vanderdonck, J. (2004). Graceful Degradation of User Interfaces as a Design Method for Multiplatform Systems. In *IUI '04 Proceedings of the 9th international conference on Intelligent user interfaces* (pp. 140-147). Funchal, Madeira, Portugal: ACM.

- Forcht, K. A., Aldridge, A., & Shorter, J. D. (2007). Implementing enterprise resource planning (ERP) for strategic competitive advantage. *Journal of Issues in Information Systems*, 8(2), 425-429.
- Groenewegen, D. M., & Visser, E. (2013). Integration of data validation and user interface concerns in a DSL for web applications. *Software & Systems Modeling*, 12(1), 35-52. <https://doi.org/10.1007/s10270-010-0173-9>
- Haines, M. N., & Goodhue, D. L. (2000). ERP Implementations: The Role of Implementation Partners and Knowledge Transfer. In *Proceedings of the Information Resources Management Association (IRMA) International Conference* (pp. 34-38). Anchorage, AK.
- Harwani, B. (2010). Form Validation. *InjQuery Recipes: A Problem-Solution Approach* (pp. 107-166). Apress.
- Henschen, D. (2013, December 12). Avon Pulls Plug On \$125 Million SAP Project-InformationWeek. Retrieved May 16, 2017, from [http://www.informationweek.com/software/enterprise-applications/avon-pulls-plug-on-\\$125-million-sap-project/d/d-id/1113061](http://www.informationweek.com/software/enterprise-applications/avon-pulls-plug-on-$125-million-sap-project/d/d-id/1113061)
- Holland, C., & Light, B. (1999). A critical success factors model for ERP implementation. *IEEE Software*, 16(3), 30-36.
- Holland, C., Light, B., & Gibson, N. (1999). A critical success factors model for enterprise resource planning implementation. In *Proceedings OfThe 7th European Conference On Information Systems* (pp. 273-287).
- Hossain, L., Patrick, J.D., & Rashid, M.A. (2001). *Enterprise Resource Planning: Global Opportunities and Challenges*. Idea Group Inc (IGI).
- ISO. (2011). ISO/IEC 25010, International Organization for Standardization, ISO, Systems and software engineering-SquaRE-Software product Quality Requirements and Evaluation-System and Software Quality Models.
- Jacobson, S., Shepherd, J., D'Aquila, M., & Carter, K. (2007). The ERP Market Sizing Report, 2006-201, 36.
- Keystone Strategy. (2007). ERP End-User Business Productivity: A Field Study of SAP & Microsoft. Keystone Strategy.
- Lambeck, C., Moller, R., Fohrholz, C., & Leyh, C. (2014). (Re-)Evaluating User Interface Aspects in ERP Systems-An Empirical User Study. In *2014 47th Hawaii International Conference on System Sciences* (pp. 396-405).
- Lengstorf, J., & Wald, K. (2016). Performing Form Validation with Regular Expressions. In J. Lengstorf & K. Wald (Eds.), *Pro PHP andjQuery* (pp. 311-342). Berkeley, CA: Apress. [https://doi.org/10.1007/978-1-4842-1230-1\\_9](https://doi.org/10.1007/978-1-4842-1230-1_9)
- Leon, A. (2008). *ERP Demystified* (Second). Tata McGraw-Hill Education.
- Leon, A. (2013). *Enterprise Resource Planning* (Third). McGraw-Hill Education.

- Lepreux, S., Vanderdonckt, J., & Michotte, B. (2007). Visual Design of User Interfaces by (De)composition. In G. Doherty & A. Blandford (Eds.), *Interactive Systems. Design, Specification, and Verification* (pp. 157-170). Springer Berlin Heidelberg.
- Li, X., & Xue, Y. (2014). A Survey on Server-side Approaches to Securing Web Applications. *ACM Comput. Surv.*, *46(4)*, 54:1-54:29.  
<https://doi.org/10.1145/2541315>
- Lykkegaard, B., & Elbak, A. (2011). IDC - Document at a Glance - LC52T. International Data Corporation (IDC).
- Marder, A. (2017, January 30). Best ERP Software 12017 Reviews of the Most Popular Systems. Retrieved May 6, 2017, from <http://www.capterra.com/enterprise-resource-planning-software/>
- Nah, F. F.-H., Zuckweiler, K. M., & Lau, J. L.-S. (2003). ERP Implementation: Chief Information Officers' Perceptions of Critical Success Factors. *International Journal of Human-Computer Interaction*, *16(1)*, 5-22.
- Ngai, E. W. T., Law, C. C. H., & Wat, F. K. T. (2008). Examining the critical success factors in the adoption of enterprise resource planning. *Computers in Industry*, *59(6)*, 548-564.
- Oja, M.-K., & Lucas, W. (2010). Evaluating the Usability of ERP Systems: What Can Critical Incidents Tell Us. In *Fifth Pre-ICIS workshop on ES Research, St. Louis*.
- Oja, M.-K., & Lucas, W. (2011). ERP Usability Issues from the User and Expert Perspectives. *Journal of Information Technology Case and Application Research*, *13(3)*, 21-40.
- Panorama Consulting Solutions. (2016). Report on ERP Systems and Enterprise Software. A Panorama Consulting Solutions Research Report.
- Paterno, F., Mancini, C., & Meniconi, S. (1997). ConcurTaskTrees: A Diagrammatic Notation for Specifying Task Models. In S. Howard, J. Hammond, & G. Lindgaard (Eds.), *Human-Computer Interaction INTERACT '97* (pp. 362-369). Boston, MA: Springer US. [https://doi.org/10.1007/978-0-387-35175-9\\_58](https://doi.org/10.1007/978-0-387-35175-9_58)
- Resig, J., Ferguson, R., & Paxton, J. (2015). JavaScript and Form Validation. In *Pro JavaScript Techniques* (Second Edition, pp. 95-105). Apress. Retrieved from [//www.apress.com/gp/book/9781430263913](http://www.apress.com/gp/book/9781430263913)
- Rosario, J. Gd. (2000). On the Leading Edge: Critical success factors in ERP implementation projects. *Business World (Philippines)*, *27*.
- SAP. (2017). SAP Business One... To Go release 9.2.
- Scaffidi, C., Myers, B., & Shaw, M. (2008). Topes: reusable abstractions for validating data. In *Proceedings of the 13th international conference on Software engineering - ICSE '08* (pp. 1-10). Leipzig, Germany: ACM Press.  
<https://doi.org/10.1145/1368088.1368090>
- Scholtz, B., Calitz, A., & Cilliers, C. (2013). Usability evaluation of a medium-sized ERP system in higher education. *Electronic Journal of Information Systems Evaluation*, *16(2)*, 148-161.

- Scholtz, B., Cilliers, C., & Calitz, A. (2010). Qualitative Techniques for Evaluating Enterprise Resource Planning (ERP) User Interfaces. In *Proceedings of the 2010 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists (SAICSIT)* (pp. 284-293). New York, USA: ACM.
- Scholtz, B., Mahmud, I., & Ramayah, T. (2016). Does Usability Matter? An Analysis of the Impact of Usability on Technology Acceptance in ERP Settings. *Interdisciplinary Journal of Information, Knowledge, and Management*, 11.
- Seffah, A., Donyae, M., Kline, R. B., & Padda, H. K. (2006). Usability Measurement and Metrics: A Consolidated Model. *Software Quality Journal*, 14(2), 159-178.
- Shang, S., & Seddon, P. B. (2000). A Comprehensive Framework for Classifying the Benefits of ERP Systems. *Americas Conference on Information Systems (AMCIS) 2000*, 39.
- Singh, A., & Wesson, J. (2009). Evaluation Criteria for Assessing the Usability of ERP Systems. In *Proceedings of the 2009 annual research conference of the South African Institute of Computer Scientists and Information Technologists* (pp. 87-95). ACM.
- Stark, P. B., & Flaherty, J. S. (2011). *The Competent Leader: 19 Critical Skills Any Manager Or Supervisor Must Know*. Human Resource Development.
- Sumner, M. (1999). Critical success factors in enterprise wide information management systems projects (pp. 297-303). ACM SIGCPR. Retrieved from <http://portal.acm.org/citation.cfm?doid=299513.299722>
- Topi, H., Lucas, W. T., & Babaian, T. (2005). Identifying Usability Issues with an ERP Implementation. In *Proceedings of the International Conference on Enterprise Information Systems (ICEIS)* (pp. 128-133).
- Vaman, J. N. (2007). *ERP in Practice: ERP Strategies for Steering Organizational Competence and Competitive Advantage*. Tata McGraw-Hill Education.
- Wortmann, J. C. (1998). Evolution of ERP Systems. In *Strategic Management of the Manufacturing Value Chain* (pp. 11-23). Springer, Boston, MA. [https://doi.org/10.1007/978-0-387-35321-0\\_2](https://doi.org/10.1007/978-0-387-35321-0_2)

## Appendix A: Questionnaires used in the Studies

### Background Information

1) Please indicate your gender.

Male  Female

2) Please indicate your age group.

Less than 21  21 to 30  31 to 40  
 41 to 50  51 to 60  Above 60

3) For how long did you use or have been using enterprise software applications or any line-of-business information system (e.g., enterprise resource planning, customer relationship management, supply chain management, accounting, etc.)?

Less than 1 Year  1 to 5 Years  6 to 10 Years  
 11 to 15 Years  16 to 20 Years  Over 20 Years

4) On average, how many hours per day have you been using such systems?

Less than 1 Hour  1 to 4 Hours  5 to 8 Hours  
 Over 8 Hours

5) How would you rate your computer skills?

Novice  Intermediate  Expert

## SAP Usability Problems

- 1) What are the main usability problems that you encounter in your daily use of SAP?

Problem 1 Description

---

Problem 1 Severity

<b>O</b>	<b>O</b>	<b>O</b>	<b>O</b>	<b>O</b>
Extremely Severe	Very Severe	Severe	Moderately Severe	Somewhat Severe

Problem 2 Description

---

Problem 2 Severity

<b>O</b>	<b>O</b>	<b>O</b>	<b>O</b>	<b>O</b>
Extremely Severe	Very Severe	Severe	Moderately Severe	Somewhat Severe

Problem 3 Description

---

Problem 3 Severity

<b>O</b>	<b>O</b>	<b>O</b>	<b>O</b>	<b>O</b>
Extremely Severe	Very Severe	Severe	Moderately Severe	Somewhat Severe



### Questions Related to Initial and Adapted SAP UIs

How does the adapted UI compare with the original SAP UI?

Usability:    Much Worse      Much Better

Comments: -----  
\_\_\_\_\_  
\_\_\_\_\_

Please answer the questions below to rate the **initial** SAP UI.

	Strongly Disagree	Strongly Agree
1) I think that I would like to use this system frequently	<input type="radio"/>	<input type="radio"/>
2) I found the system unnecessarily complex	<input type="radio"/>	<input type="radio"/>
3) I thought the system was easy to use	<input type="radio"/>	<input type="radio"/>
4) I think that I would need the support of a technical person to be able to use this system	<input type="radio"/>	<input type="radio"/>
5) I found the various functions in this system were well integrated	<input type="radio"/>	<input type="radio"/>
6) I thought there was too much inconsistency in this system	<input type="radio"/>	<input type="radio"/>
7) I would imagine that most people would learn to use this system very quickly	<input type="radio"/>	<input type="radio"/>
8) I found the system very cumbersome to use	<input type="radio"/>	<input type="radio"/>
9) I felt very confident using the system	<input type="radio"/>	<input type="radio"/>
10) I needed to learn a lot of things before I could get going with this system	<input type="radio"/>	<input type="radio"/>

Please answer the questions below to rate the **adapted** UI.

	Strongly Disagree	Strongly Agree
1) I think that I would like to use this system frequently	<input type="radio"/>	<input type="radio"/>
2) I found the system unnecessarily complex	<input type="radio"/>	<input type="radio"/>
3) I thought the system was easy to use	<input type="radio"/>	<input type="radio"/>
4) I think that I would need the support of a technical person to be able to use this system	<input type="radio"/>	<input type="radio"/>
5) I found the various functions in this system were well integrated	<input type="radio"/>	<input type="radio"/>
6) I thought there was too much inconsistency in this system	<input type="radio"/>	<input type="radio"/>
7) I would imagine that most people would learn to use this system very quickly	<input type="radio"/>	<input type="radio"/>
8) I found the system very cumbersome to use	<input type="radio"/>	<input type="radio"/>
9) I felt very confident using the system	<input type="radio"/>	<input type="radio"/>
10) I needed to learn a lot of things before I could get going with this system	<input type="radio"/>	<input type="radio"/>

Please choose **three terms** that you think mostly describe the **initial SAP UI**.  
The choice could be positive, negative, or a combination of both.

Positive	Negative
<input type="radio"/> Clear	<input type="radio"/> Annoying
<input type="checkbox"/> Easy to use	<input type="checkbox"/> Complex
<input type="checkbox"/> Effective	<input type="checkbox"/> Confusing
<input type="checkbox"/> Efficient	<input type="radio"/> Difficult
<input type="checkbox"/> Effortless	<input type="radio"/> Disruptive
<input type="checkbox"/> Fast	<input type="radio"/> Frustrating
<input type="checkbox"/> Friendly	<input type="radio"/> Hard to Use
<input type="checkbox"/> Straight Forward	<input type="radio"/> Time-consuming
<input type="checkbox"/> Time-Saving	<input type="radio"/> Stressful
<input type="checkbox"/> Understandable	<input type="checkbox"/> Overwhelming

Please choose **three terms** that you think mostly describe the **adapted UI**.  
The choice could be positive, negative, or a combination of both.

Positive	Negative
D Clear	D Annoying
D Easy to use	D Complex
D Effective	D Confusing
D Efficient	D Difficult
D Effortless	D Disruptive
D Fast	D Frustrating
D Friendly	D Hard to Use
D Straight Forward	D Time-consuming
D Time-Saving	D Stressful
D Understandable	D Overwhelming

## Appendix 8: Identified Error Validation Types

<b>Error Message</b>	<b>Validation Type</b>
<b>Users - Setup UI</b>	
This entry already exists in the following tables 'Users' (OUSR) (ODBC-2035) [Message 131-183]	Exists
<b>Posting Periods UI</b>	
Period is overlapping with an existing period. [Message 636-20]	Exists
You cannot remove this item; its transactions are linked to an object, '2017-03' [Message 173-44]	Exists
<b>Document Numbering Series - setup UI</b>	
Duplicate series name [Message 173-124]	Exists
Series overlaps object A/R Invoices	Exists
<b>Business Partner UI</b>	
You cannot remove a BP with a balance not equal to 0	Comparison
You cannot remove BP; linked Sales Orders exist	Exists
Customer Currency Should be USD	If / Comparison
This Business Partner should be under dealer group. It's already used as Dealer in applications – SO	Exists / Comparison
Sales Employee should be set as [-No Sales Employee-]	Comparison
You have to define if this BP is a Personal Account or not	Mandatory
Vendors BP should be set as non-Personal Account	Comparison
This BP should be under DSP group. It's already used as Accrual DSP in Items Groups	Exists / Comparison
This Business Partner should be under DSP group. It's already used as DSP in Applications – SO	Exists / Comparison
You cannot delete BP used as Dealer in applications - SO	Exists
You cannot delete BP used as Bill To in the list of Bill To	Exists
You cannot delete BP used as Bill To in applications - SO	Exists
You cannot delete BP used as Accrual DSP in Items Groups	Exists
You cannot delete BP used as DSP in applications - SO	Exists
You cannot delete BP already exists in Acc Transactions	Exists
Tel 1 should be 8 numbers	Restrict Type and Size
Tel 2 should be 8 numbers	Restrict Type and Size
Mobile phone should be 8 numbers	Restrict Type and Size
Address phone1 should be 8 numbers	Restrict Type and Size
You are not allowed to change the BP Name	Exists
BP series and type are not matching - can be either leads/lead or clients/customer	Comparison
Missing BP name	Mandatory
Fiscal/VAT-Ref is mandatory when account is not personal	Comparison

Error Message	Validation Type
<b>Chart of Account UI</b>	
You have to define the currency of the cash account in UDF	Mandatory
the G/L Account should have the format as [NNNNN.NNNNN]	Regular Expression
Accounts with related VATBP in Accounting Transactions cannot be set as not [VAT BP]	Exists / Comparison
<b>Outgoing axment 4J;</b>	
Total amount should not be negative	Comparison
<b>A/P Invoice EWt</b>	
Document Numbering cannot be Manual	Comparison
Wrong Numbering Series Definition- Type	Exists
The Supplier Control Account is not defined in the BP	Exists
Wrong Document numbering definitions -Category	Exists
Supplier Control Account doesn't match the one defined in the document numbering	Exists
The selected Items are related to items groups that don't match the definition. Category in Document numbering is<> than the item group Numbering Series Category	Exists
The selected Items are related to items groups that have multiple document series Groups	Exists
The selected Items are related to items groups that don't match the definition	Exists
<b>A/P Credit Memo.;! l. .. !!</b>	
Document Numbering cannot be Manual	Comparison
Wrong Numbering Series Type	Exists
The Supplier Control Account is not defined in the BP	Exists
Wrong Document numbering Definitions- Category	Exists
Supplier Control Account doesn't match the one defined in the document numbering	Exists
The selected Items are related to items groups that don't match the definition	Exists
The selected Items are related to items groups that have multiple document series Groups	Exists
The selected Items are related to items groups that don't match the definition	Exists
<b>PurchaseOrders UI</b>	
Document Numbering cannot be Manual	Comparison
Wrong Numbering Series Definition- Type	Exists
The Supplier Control Account is not defined in the BP	Exists
Wrong Document numbering Definitions- Category	Exists
The selected Items are related to items groups that don't match the definition	Exists

<b>Error Message</b>	<b>Validation Type</b>
<b>Goods Receipt PO UI</b>	
Document Numbering cannot be Manual	Comparison
Wrong Numbering Series Definition- Type	Exists
The Supplier Control Account is not defined in the BP	Exists
Wrong Document numbering Definitions- Category	Exists
The selected Items are related to items groups that don't match the definition	Exists
The selected Items are related to items groups that have multiple document series Groups	Exists
The selected Items are related to items groups that don't match the definition	Exists
<b>Go'ods Return UI</b>	
Document Numbering cannot be Manual	Comparison
Wrong Numbering Series Definition- Type	Exists
The Supplier Control Account is not defined in the BP	Exists
Wrong Document numbering Definitions -Category	Exists
The selected Items are related to items groups that don't match the definition	Exists
The selected Items are related to items groups that have multiple document series Groups	Exists
The selected Items are related to items groups that don't match the definition	Exists
<b>Incoming Payment UI</b>	
You cannot add payment having postdated check	Comparison
Wrong Cash Account	Exists
Wrong Check Account	Exists
Payment on Account cannot be Negative!	Comparison
<b>Exchange Rate Differences 01</b>	
Offset account is missing. Field: Exch. Rate Gain Acct (AIR)	Mandatory
Invalid date. Field: Execution Date	Mandatory
Customer BP is inactive	Exists / Comparison
<b>Connection UI</b>	
Adjust Customer Code - BP code is not a customer	Exists
You don't have authorization to add a connection manually	Exists / Comparison
Customer Code does not exist	Exists Item
Code does not exist	Exists
Missing Item Code	Mandatory
Connections with this Link Type cannot have discount	If / Comparison
You selected the wrong Document Series for this Category	Comparison
The Connection Status for CANCELLED Item Code should be CANCELLED	If / Comparison



<b>Error Message</b>	<b>Validation Type</b>
<b>Payment Method UI</b>	
Adjust BiiiTo Code- BP code is not a customer	Exists
Wrong Code	Exists
Wrong Name	Comparison
Discount% should be between 0 and 100	Range
Customer Code does not exist	Exists
Total Pay share should be 100%	Sum
Total Pay share should be the same as Selling Price	Comparison
Connections with this Link Type cannot have discount	Exists / Comparison
You have to update the discount type	If / Comparison
Discount Type should be equal to-	If / Comparison
Discount Amount should >0 and <= Price	Range
Selling price is different than the sum of internet price- discount	Sum / Comparison
Payment Method is mandatory	Mandatory
You have to specify the Bank	Mandatory
Delivery Method is mandatory	Mandatory
You have to specify the Account No or the IBAN	Mandatory
You have to specify the Credit Card Expiry Date	Mandatory
You Have to define the Currency	Mandatory
You cannot set Discount Type as Amount for this type of Connections	Exists / Comparison
You cannot split the payment method of this type of connections to more than 1 Customer	Count
<b>A/R Invoice UI</b>	
Wrong Numbering Series Type	If / Exists
Choose another document Series	If / Exists
Wrong VAT Group	If / Exists
Wrong VAT Amount	If / Exists
Wrong Service Cost Center (Dim1) By Item	If / Exists
Wrong Package Cost Center (Dim2) By Item	If / Exists
Missing Cost Center (Dim2)- Sales Invoice	Mandatory
Missing Cost Center (Dim2)- Debit Note	Mandatory
You cannot enter discount for documents containing this Item	Comparison
Connection is mandatory for one or more items	Exists / Mandatory
You cannot assign more than one connection per invoice	If / Count Distinct
Wrong Purchase Accrual Definition (VAT Group)	If / Exists
Wrong Purchase Accrual Definition (Cost Centers)	If / Exists
Wrong Purchase Accrual Definition (Accrual Expense)	If / Exists
Wrong Purchase Accrual Definition (Accrual Liability)	If / Exists
Wrong Purchase Accrual Definition (Accrual: Expense Cost)	If / Exists
Wrong Purchase Accrual Definition (Accrual: Accrual Cost)	If / Exists
The entered DSP Fee is unacceptable	If / Exists
DSP Fee and DSP Currency Should be Null for Accrual Groups by Item	If / Exists

Error Message	Validation Type
A/R Invoice UI	
Wrong DSP Assignment	If / Exists
Wrong document series	If / Exists
Discount should not be negative	Comparison
A/R Credit Memo UI	
Wrong Numbering Series Type	If / Exists
Choose another document Series	If / Exists
Wrong VAT Group	If / Exists
Wrong VAT Amount	If / Exists
Wrong Service Cost Center (Dim1)By Item	If / Exists
Wrong Package Cost Center (Dim2) By Item	If / Exists
Missing Cost Center (Dim2)- Return	Mandatory
Missing Cost Center (Dim2)- Credit Note	Mandatory
You cannot enter discount for documents containing this Item	Comparison
Connection is mandatory for one or more items	Exists / Mandatory
you cannot Assign more than one connection per document	If / Count Distinct
Wrong Purchase Accrual Definition (VAT Group)	If / Exists
Wrong Purchase Accrual Definition (Cost Centers)	If / Exists
Wrong Purchase Accrual Definition (Accrual Expense)	If / Exists
Wrong Purchase Accrual Definition (Accrual Liability)	If / Exists
Wrong Purchase Accrual Definition (Accrual: Expense Cost)	If / Exists
Wrong Purchase Accrual Definition (Accrual: Accrual Cost)	If / Exists
The entered DSP Fee is unacceptable	If / Exists
DSP Fee and DSP Currency Should be Null for Accrual Groups by Item	If / Exists
Sales BOM components cannot be deleted	Exists
Individual sales BOM components cannot be duplicated separately	Exists
Wrong VAT Group	If / Exists
Wrong VAT Amount	If / Exists
Wrong Service Cost Center (Dim1)By Item	If / Exists
Wrong Package Cost Center (Dim2) By Item	If / Exists
You cannot enter discount for documents containing this Item	Comparison
DSL Number should be 8 digits only	Restrict Type and Size
DSL Number cannot start with 3	Regular Expression
DSL Number does not exist in the list of POPs	Exists
You have to set a Value to the POP	Mandatory
Phone Number exists in multiple POPs	Exists
There is mismatching between DSL Number and selected POP	Exists
You cannot order more than one modem in one SO	Exists / Count
Item and document series are not matching	Exists
Duplicate DSL Number	Exists
Missing Plan	Mandatory

Error Message	Validation Type
<b>Sales Order UI</b>	
Missing Delivery Method	Mandatory
Missing Mobile Number	Mandatory
Missing First name, Middle name, or Last name	Mandatory
Missing Payment Method	Mandatory
Missing Bank or IBAN or Account Holder Name or Account No. or Credit Card Expiry Date	Mandatory
IBAN should be 28 characters	Restrict Type and Size
Wrong Bank	If / Comparison
Account No. should be 16 digits without space, -, and /	Restrict Type and Size / Regular Expression
Plan inactive	Exists
Dealer inactive	Exists
No change in plan	Exists
Duplicate Sales Order	Exists
Multiple Changes for Plan not allowed in the same month	Exists / Count / Range
You are not allowed to add changes starting 27 of the month	Comparison / Range
Missing reference	Mandatory
Missing BP name	Mandatory
There is a mismatch between the chosen offers	Comparison
Wrong plan with the chosen offer	If / Comparison
Payment method should be Prepaid Card with this offer	If / Comparison
You are not allowed to add changes after December 18	Comparison / Range
Changes for only 1 connection can be added per sales order	Count Distinct
Discount should not be negative	Comparison
Wrong DSL plan with the chosen offer	If / Comparison
These items should not have a discount	If / Comparison
Missing DSL parental control	Mandatory
Missing DSP name	Mandatory
Mismatching between plan and DSP	Comparison
Missing address	Mandatory
Missing discount type for the defined discount amount	Mandatory
You are not allowed to change the application ref.	Comparison
Mobile number should be 8 digits only;it cannot contain space,-, or /	Restrict Type and Size / Regular Expression
Invalid email- if you do not have the customer email, leave it empty	Regular Expression
ID is mandatory- it should not contain spaces and special characters	Mandatory / Regular Expression
Missing Email	Mandatory
New email field should be filled with Yes or No	Restrict Value

<b>Error Message</b>	<b>Validation Type</b>
<b>Journal entry UI</b>	
Unbalanced Transaction [Message 131-91]	Comparison
You have to select manual JE series	Comparison
Document Series does not match the transaction code	Exists
You cannot post transactions in January 1st	Comparison
You cannot remove item; linked transactions exist	Exists
Item code already exists [Message 3504-19]	Exists
You have to define an Item Group	Mandatory
Item Sales/Purchase VAT Group differ from related items groups Sales/Purchase VAT Groups	Exists
Item Evaluation Method should be the same as the item group	Exists
Standard Item Cost should be 0	Comparison
[Set G/L Accounts by] Value should be [Item Group]	Comparison
Blank Service Group (Dim1) / Service type (Dim2)	Mandatory
Blank Service Group Code (Dim1) / Service type Code (Dim2)	Mandatory
Service Group Code (Dim1) not exist	Exists
Service Type Code (Dim2) not exist	Exists
Item used should not be a Purchase item but an Inventory Item	If / Comparison
Accrual Cost and Currency should be defined	Mandatory
Accrual Cost Type should be defined	Mandatory
Mismatching between Accrual Cost and Accrual Cost Type	Comparison
Items related to Groups with enabled Purchase Accrual cannot be Inventory nor Purchase items	If / Exists
You cannot create an item related to an Item Group without Expense Account	If / Comparison
You cannot create an item related to an Item Group without defined series categories	If / Comparison
Connection Mandatory in SI/SR should be Y or N	Restrict Value
Connection Mandatory in SO should be Y or N	Restrict Value
Itemcode can contain maximum 20 characters	Restrict Type and Size
Item series and asset class do not match	Exists
Wrong item group	Exists
Item group and asset class do not match	Exists