# USER ACCEPTANCE TESTING

# USER ACCEPTANCE TESTING

# CONTENT

# APPENDICES



# PICTURES



# FIGURES

# TABLES

# LIST OF ABBREVIATIONS (OR) SYMBOLS

| | |
|---|---|
| AOS | Application Object Server |
| AIF | Application Integration Framework |
| ATDD | Automated Test Driven Development |
| BI | Business Intelligence |
| BOM | Bill of Materials |
| CEO | Chief Executive Officer |
| COO | Chief Operating Officer |
| CRM | Customer Relationship Management |
| DGPS | Differential Global Positioning System |
| EP | Enterprise Portal |
| ERP | Enterprise Resource Planning |
| GNSS | Global Navigation Satellite System |
| GPRS | General Packet Radio Service |
| GSM | Global System for Mobile Communication |
| HRM | Human Resources Management |
| IEEE | Institute of Electrical and Electronics Engineers |
| ISO | International Standards Organization |
| IT | Information Technology |
| IWM | Inventory and Warehouse Management |
| LLWAS | Low Level Winds hear Alert System |
| PDCA | Plan-Do-Check-Act |
| POC | Proof Of Concept |
| R&D | Research and development |
| RF | Radio Frequency |
| SAN | Storage Area Network |
| SAP | Systems, Applications, Products |
| SCADA | Supervisory Control and Data Acquisition |

| | |
|---|---|
| SDLC | Software Development Life Cycle |
| SME | Small and Medium-sized Enterprise |
| SMD | Surface Mount Device |
| TDD | Test Driven Development |
| UAT | User Acceptance Testing |
| UHF | Ultra High Frequency |
| UPS | Uninterrupted Power Supply |
| VHF | Very High Frequency |
| XML | eXtensible Markup Language |

# 1 INTRODUCTION

## Research problem

In order to complete the ERP project successfully SATEL Oy (later SATEL) needs the user acceptance test plan. The main research difficulty of this thesis was to design the user acceptance testing plan for ERP implementation. Currently SATEL does not have the test plan. This thesis concentrates on designing a plan based on the theory of the UAT. The plan is designed by researching user acceptance testing from literature, articles and the Internet. The plan will be designed based on the theory found in the literature.

There are limitations in researching this domain area. It has not been widely researched. There is very little written literature on the subject, only 3-4 books can be found concentrating only on UAT.

The lack of literature about the UAT has been challenging. It has been difficult to collect enough references. To avoid this limitation, different types of methods have been used to gain more information on the subject. The Internet and social media platforms such as Slideshare, LinkedIn and Twitter have played an important role in acquiring information, direct contacts to the UAT professionals have been made by e-mails and phone calls. How the information was collected can be seen in appendix 1. I contacted, via e-mail, Mr. James Windle who has years of experience in the field of the UAT. Another challenge has been terminology. Testing terminology is not universally agreed and it is very difficult to differentiate terms in the testing domain that applies to the UAT as well (BS-7925-1 2010).

## The research method and justification

The research method used in this thesis is a constructive research. The objective of constructive research is to find a solution for a real world practical prob-

lem. Constructive research is based on a theoretical body of knowledge, literature and information from many sources such as the Internet, articles, journals and personal communication. (Lassenius et al. 2001, 3-5.)

Constructive research is widely used in software engineering and the most used methodology in computer sciences. Research is normally divided into basic or applied research. Constructive research is applied research. The challenge in this method is that must describe well how the plan is implemented in reality. (Järvinen 2001, 88.) Constructive research methodology is interesting and important in the field of software development (Nummenmaa 2007, 2).

Constructive research can be used when designing a new construction or a plan from theoretical framework. The objective is to achieve a practical solution to a relevant problem.

Constructive research process consists of finding practically relevant problem and understanding the topic by reading the relevant literature about the subject (Seppänen 2004, 10). The key idea in this methodology is construction of new – in this case, user acceptance testing plan. (UAT) (Crankovic 2010, 2).

The issue is how to design a user acceptance testing plan. It is the symptom that acts as a construction. It has a practical relevance related to the issue that SATEL needs a user acceptance plan. It is based on empirical knowledge of creation that offers final benefits as a complete user testing plan. The theory connection refers to multiple references such as books, articles, web-links, and interviews with the UAT professionals.

The four elements in a constructive research method are to find practically relevant problem, which has a scientific potential. The general and comprehensive knowledge of the subject needs to be obtained and after that a plan, an idea or a product can be constructed. After the construction is done the demonstration with the solution implementation is essential. In addition, theoretical connections and research contribution of the solution should be illustrated. Finally, the applicability of the solution should be examined. (Kasanen et al. 1993, 246.) The building blocks of the constructive research can be seen in below figure 1.

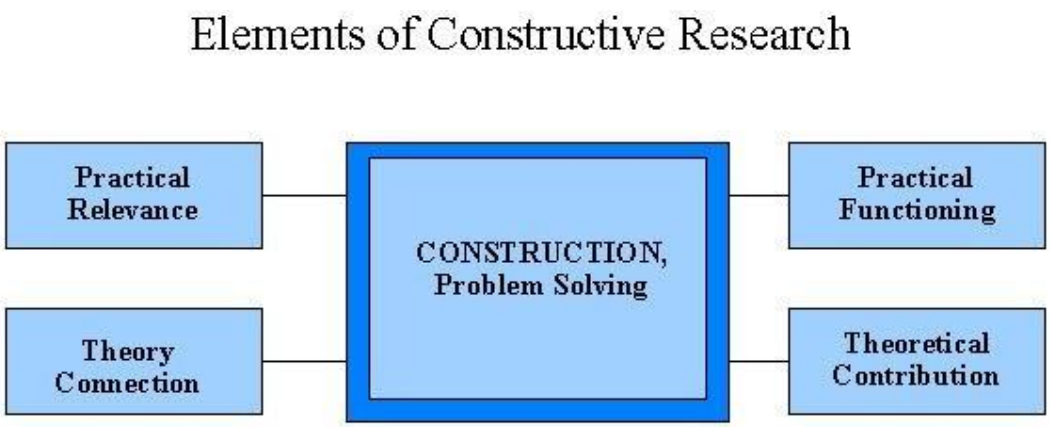



Figure 1. Elements of constructive research. (Kasanen et al. 1993, 246).

This research method poses the challenges of how to prove that the actual user acceptance testing plan works in SATEL. The applicability of the solution to SATEL is an objective as well. The main research paradigm is constructivism with axiological assumption. Research approach is inductive and data collection technique is qualitative.

## SATEL Oy

SATEL was founded in 1986 by Pekka Aura and two other radio engineers. SATEL is a privately owned company by PASATEL Oy. SATEL is a Finnish electronics and telecommunications company that specializes in the design, manufacturing and global sales and marketing of radio modems for wireless data communication and alarm transfer. SATEL is one of the leading radio modem suppliers in the world, the biggest in Europe, operating through a wide distributor network. They have 70 distributors and that cover over 100 countries. They manufacture about 35 000 radio modems in a year. SATEL headquarters are located in Salo. (SATEL Oy 2013)


Picture 1. SATEL Oy headquarters are located in Salo. (SATEL Oy 2007).

Head quartered in Salo is management, R&D, production and marketing. SATEL's headquarters can be seen in picture 1. SATEL employs 67 professionals in total. Employees divided into different organizational levels can be seen in below figure 2.

Company employs totally **67** professionals
1. Administration, finances, IT: **5** persons
2. Sales and marketing and technical support: **12** persons
3. R&D: **29** persons
4. Production and purchasing: **21**persons

Figure 2. An employee statistics. (SATEL Oy 2012a).

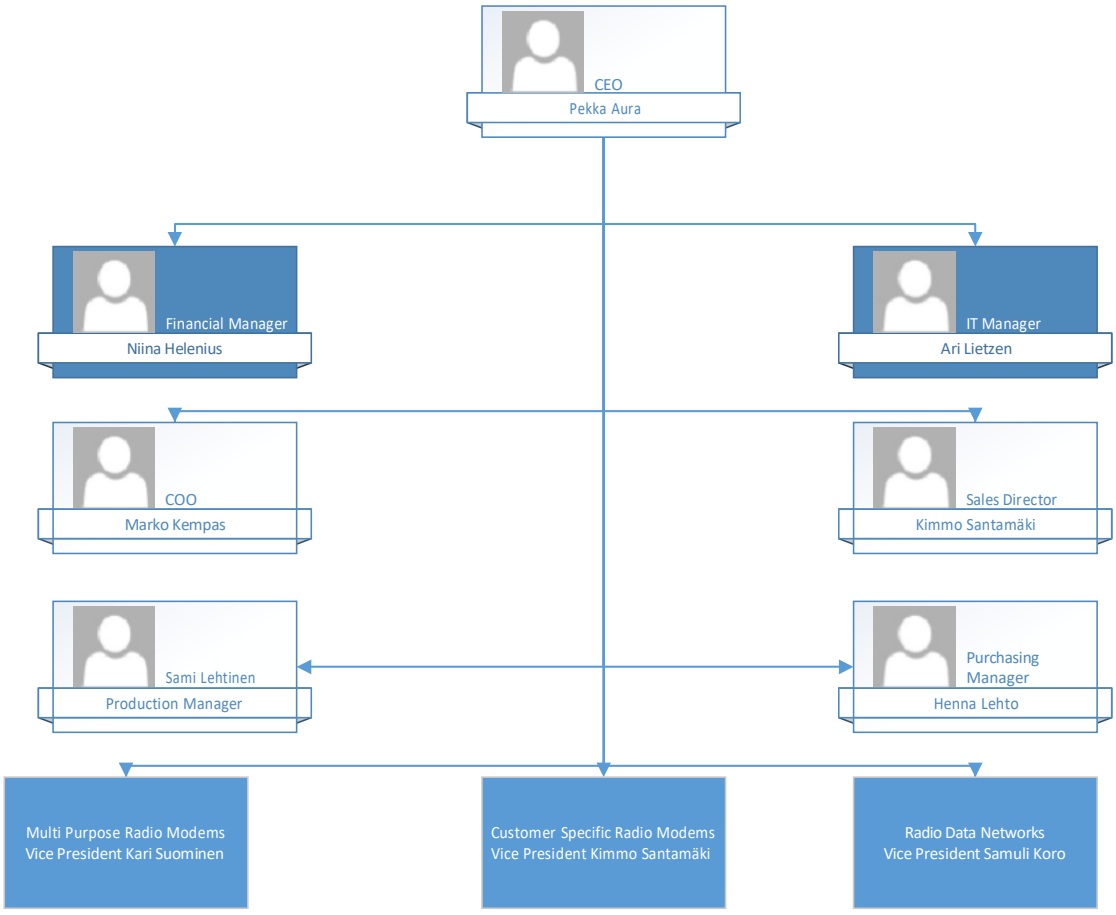



Figure 3. SATEL Oy's organizational chart.

The chart presented in figure 3 describes our executive level organizational chart. IT and Finance are supportive organizations to the company and they operate under the direct command of the CEO. The CEO, COO, Production manager and Purchasing manager form the executive management of the company known as the management group. Each business unit has a management of its own.

The operation is divided into three business units. The turnover is around 13 – 15 million euros and 90 % of the turnover comes from export. Most of the sales come from Europe, North and South America and Asia. (Company presentation, 2012a.) This can be seen in below figure 4.

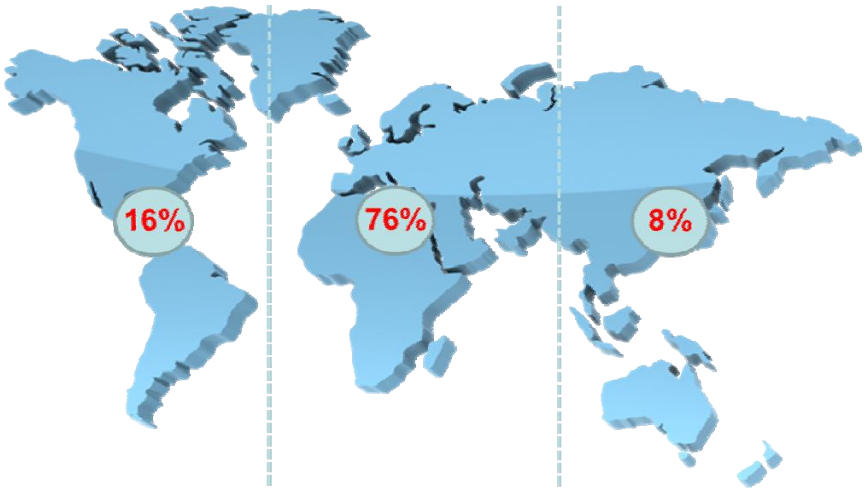



Figure 4. Sales geographically in 2011. (SATEL Oy 2012b).

Our goal is satisfied customers. We strive to achieve only the best service and highest quality products. We are continuously developing our products to meet market changes and demands. They value healthy business with continuous profitable growth while providing a safe work community and enthusiastic personnel.

Our financial situation is in excellent shape allowing us easily to have AAA credit rating. We follow one principle “We cannot spend more than we earn.” We invest more than 10% of our annual turnover in research and development. (SATEL 2012c.)

Our mission is to provide wireless communication solutions to improve customer’s business. Our vision is to be the most sought after radio network supplier globally.

## Radio modem products

The main product line of SATEL, is radio modems. They are divided into these following groups smart radio modems, classic radio modems and customer specific radio modems. A radio modem is used for transmitting data several kilometers over a wireless connection to another radio modem over point-to-point or –multipoint link. A radio modem network is a private network and with no operator costs associated with transferring the data. In addition, radio modem networks can use licensed or unlicensed UHF or VHF frequency bands. In table 1 below comparison between radio modems and competitive technologies can be seen.

Table 1. Differences between competitive technologies (SATEL Oy 2012c)

| Feature \ Tech. | 2.4 GHz Spread Spectrum | GSM / GPRS | Narrow-Band Radios (SATELLINE) |
|---|---|---|---|
| Setup of Communication | Continuous, Packet based comm. | Dial-up for comm. Circuit sw (GSM), Packet (GPRS) | Continuous |
| Transmission Delay | 20 – 100 ms | Seconds | 20 – 30 ms |
| Data Interface | RS-232 | RS-232 | RS-232 / RS-485 (RS-422) |
| Radio Coverage | 100 – 500 m | Due to network coverage of the mobile operator | 10 – 50 km |
| Data Rate | up to 115 kbps | 9600 – 38400 bps | 9600 / 19200 bps |
| Communication Reliability | Not very reliable due to congested RF band | GSM: Reliable GPRS: Delays and data rate changes acc. to network load. | Reliable, high immunity to out-of-channel interferences |

The operational range of a radio modem varies depending on transmission power, antenna gain, mast height, and environment. Radio modems can be used in SCADA, factory automation, traffic, GNSS, security, and telemetry applications. A few of typical applications for a radio modems can be seen in below table 2.

Table 2. Application examples for a radio modem. (Short Form Catalogue 2012d).

| Application | Application example |
|---|---|
| **Traffic** | Traffic light priority, parking house displays, passenger information systems |
| **SCADA (water, electricity, gas, oil, heat)** | Clean/waste water systems, smart grids, oil/gas pipeline control and AMR (automated meter reading) |
| **DGPS Land Survey** | Construction surveying, machine control, construction monitoring |
| **Automation** | Automated guided vehicles, process control, snow gun remote control, irrigation systems, M2M |
| **Fleet management** | Railways, airports, emergency vehicles |
| **Sport score service** | Athletics, orienteering, car / boat racing |

**Example: How to use a radio modem**

SATEL radio modems are used at airports. In Bangkok at their local airport, radio modems handle a LLWAS-system (Low Level Wind shear alert system). The system observes inconsistencies in the winds surrounding the airport. Sudden wind shears or downdrafts are dangerous to aircraft during take-off or landing. The electricity required for the masts (provided by Vaisala) is obtained from solar panels and the data transfer is handled by SATEL radio modems. (SATEL Success Stories, 2012e.)

### Introduction of the business units

Operations at SATEL were divided into business units in 2011. Goals of the business units are:

- Efficient management
- Careful selection of projects based on revenue, project management and scheduling of large volume shipments
- Increasing turnover
- Constant development of the product lines.

These are the basic goals and principles for the business units at SATEL. In the next chapters activities of the business units are described in further detail. (SATEL company note 2011.)

### Radio Modems

The radio modems business unit is responsible for radio modems. The objective of this business unit is to make sure that radio modem products are being updated while having the highest quality. The second objective is to develop business to increase turnover. The next objective is to update and develop marketing materials which include brochures, web, user guides and other publications related to radio modems. Finally, the radio modems business unit takes care of the customer feedback in co-operation with the marketing department. (SATEL Oy 2012f.)

### Customer specific radio modems

The customer specific radio modems unit is responsible for OEM business. They also must acquire new key customer groups and increase overall sales of the company.

## Radio Data Networks Solutions

This business unit takes care of large turnkey projects such as BESCOM which is worth 10 million USD. This project is in India and it is the biggest project in the company's history. We are providing a data communication system to Bangalore city's electricity infrastructure. SATEL Radio Data Network Solutions business unit has two main functions, first to ensure the success of the Indian project and secondly, to identify other projects such as this Indian project globally.

# 2 ERP IMPLEMENTATION

The ERP implementation phases

The ERP implementation consists of few phases. The ERP project begins from the customer specifying the requirement. The next phase is design and development done by the developers. After this phase the customer carries out user acceptance testing. The last phase of the ERP implementation is deployment and maintenance. This phase will be carried out together with the customer and the ERP consultants.

The ERP implementation methodologies differ from each other in different ERP systems. In figures 5 and 6 below are two examples from Microsoft and SAP. In my opinion, although different methodologies have their own approaches, they will use their own terminology for implementation. The objective is the same in both approaches, to have implementation successfully completed.


Figure 5. The ERP implementation phases in Microsoft's Sure Step methodology. (Microsoft 2010, 1).

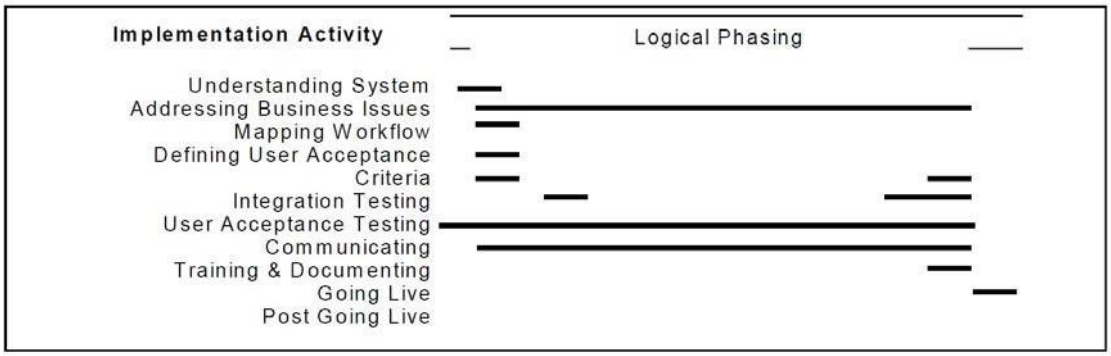



Figure 6. SAP Implementation phases. (Kawalek & Harper 2002, 17).

It is ideal to have user involvement in nearly all phases of the ERP project. This would indicate that every effort has made in the diagnostic, analysis and design phases of the ERP project. (Kawalek & Harper 2002, 14.) These early phases require much user participation.

This thesis is focusing on user acceptance testing but it is important to know the different phases of the ERP project. User participation in the ERP implementation is key to a successful ERP project. (Kawalek & Harper 2002, 19.) The user acceptance testing is a vital phase before deployment of the new ERP.

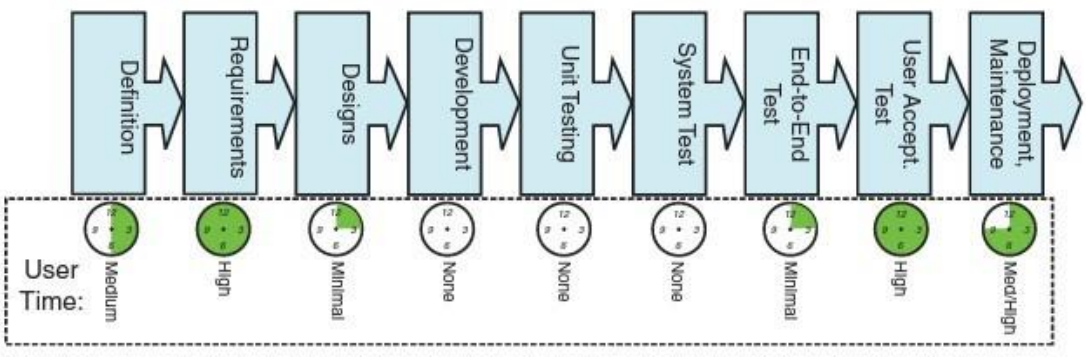



Figure 7. An Ideal user involvement in phases of the ERP project. (Cimberman 2007, 10).

Figure 7 above outlines an ideal user participation in the ERP project from definition to deployment and maintenance. The figure informs that the user acceptance testing requires high user participation from the start of the ERP project to the end. (Cimberman 2007, 10.)

## Reasons for ERP implementation

There are many reasons for organizations to implement the ERP project. Changes or growth of business, changes in external environment, escalating weaknesses of the current system or withdrawal of hardware and system software support (Windle 2010, 11).

ERP software has a certain lifecycle. A system is born, matures, degrades and dies. This is called the lifecycle of an IT system. Figure 8 below illustrates the lifecycle of an IT system (Windle 2010, 11).

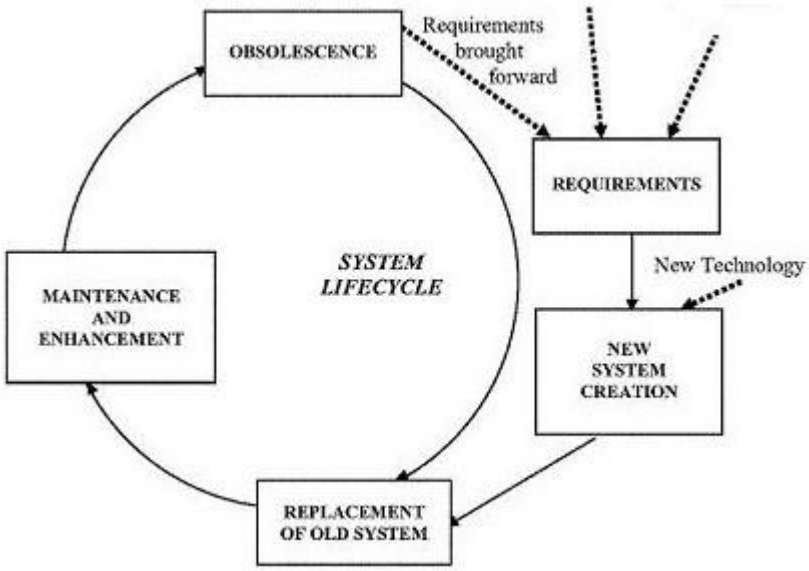



Figure 8. The ERP lifecycle. (Windle 2010, 11).

The ERP project begins when a business considers the viability, costs and benefits of a replacement system. This phase usually includes comparison between different types of ERP systems on the market and their suitability for the organization.

Fit-gap analysis or feasibility study determines whether a project is viable or not. It considers alternative ways of meeting requirements. In commercial cases such as ERP implementation, the return on investment plays an important role. (Windle 2010, 13.)

Before the ERP software is chosen, fulfilling the organizations definition and requirements, analysis is needed. The user-level involvement is usually high in the requirement analysis phase (Cimberman 2007, 9). After requirement analysis is done ERP supplier starts designing and coding phases where the requirements are turned into functionalities in a new ERP system. This can be seen in figure 9.

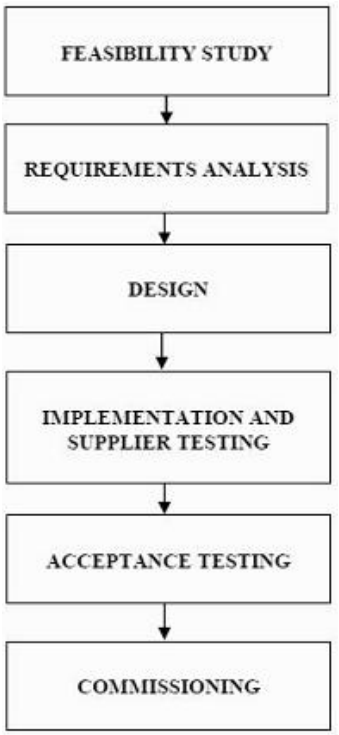



Figure 9. The steps in the ERP -software development. (Windle 2010, 12).

ERP supplier develops the system according to the customer requirements. They also provide various types of testing such as unit testing, integration testing and system testing. After a supplier has developed the system to a testable level UAT can be done. After a successful UAT deployment starts. This can be seen in figure 10 below.

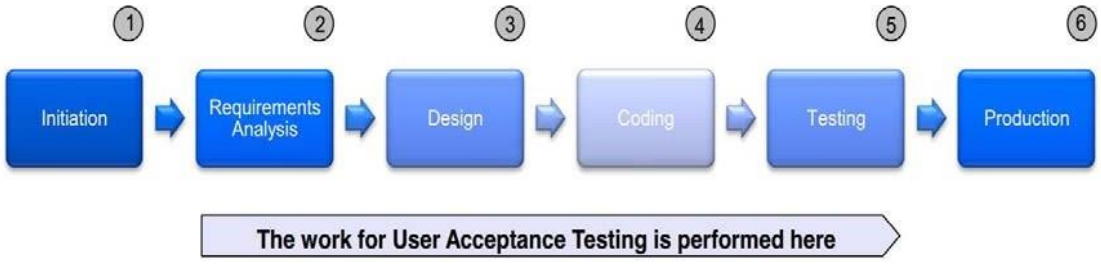



Figure 10. The UAT timing during the ERP implementation project. (Bordo 2010, slide 8).

The customer verifies that the requirements are met and the system is working accordingly. Actually the UAT is performed from the initiation to the deployment of the ERP system.

## Success factors and challenges in the ERP implementation

There are many challenges in the ERP implementation project. The ERP implementation usually starts from the business requirements or the current system is obsolete and needs to be replaced by a new system.

One of the success factors of the ERP implementation project is management's commitment and support. The ERP implementations require key business process owners to have a clear vision of how the business functions is conducted in the company in order to serve and satisfy the customers. Abbaszadeh et al. 2010, 82.)

The main objective in the ERP implementation is to change management of day to day business. The business processes should be changed that in a way that fits the new ERP system at cost saving growth to the company. (Abbaszadeh et al. 2010, 82.)

It's not always wise to try to fit every business process in the new ERP if it requires tremendous tailoring. Tailoring raises the cost of the ERP software and

increases maintenance costs as well. On the other hand some tailored features could be better than what the new ERP provides. Tailoring could bring competitive benefits over the competitors and tailor-made features could be better than generic ERP functionality (Kawalek & Harper 2002).

Re-engineering business processes is a key element for measuring benefits between generic ERP processes and tailor-made functionalities. Re-engineering business processes focuses in gaining significant improvements in cost, quality and service (Abbaszadeh et al. 2010, 83).

Using a consultant during the ERP implementation could be a good option. They may have required experiences in specific field (Abbaszadeh et al. 2010, 83). However the business owners may have required expertise in specific field, so that there is no need for business consultants at all (Kawalek & Harper 2002).

Communication is very important between business and IT personnel. This is the most challenging and difficult task in the ERP implementation (Abbaszadeh et al. 2010, 83).

Project management and training are crucial success factors. Project management oversees scheduling coordination, and use of skills and knowledge. Project management ensures that objectives during implementation are achieved by monitoring all activities. Successful ERP implementation requires excellent project management. Everyone who uses the ERP system needs to be well trained. Training will increase the success of the ERP implementation (Abbaszadeh et al. 2010, 83). The UAT is part of the crucial success factors in any ERP implementation project as well.

# 3 DESIGING THE USER ACCEPTANCE TESTING

## What is testing?

IEEE90 defines the testing with following paragraph:

"The process of operating a system or component under specified conditions, observing or recording the results, and making an evaluation of some aspect of the system or component" (IEEE90 1990, 76).

Testing will ensure that the target of test works as expected and it will ensure quality of the software. The most important item for successful testing is the provider's willingness to see defects. Testing is an investment that brings quality knowledge (Pyhäjärvi 2012, 8).

Buildings up complex systems require various people. Individually they don't have some information, but collectively enough for a full system. Testing provides important feedback, because defects transferred to production without testing can be very expensive to fix later. The company could lose its reputation with such defects. (Pyhäjärvi 2012, 9.) Testing is the most critical phase in an ERP implementation project and only few are planning it properly (Niemi 2006, 28).

The objective during the software development is to produce high quality software which only comes by testing. Most of the test scenarios during this phase are done from the developer's viewpoint. The UAT differs from typical test scenarios by putting the user in charge and by doing the UAT it is possible to have the user's point of view in the software quality. (Leung et al. 1997, 137.)

## What is user acceptance testing?

This focuses only on user acceptance testing (UAT). It plays a key role when the customer accepts a new system "going live". Without the UAT the risk that

the ERP implementation could fail is high. Therefore it is important to do the user acceptance testing according to a detailed plan.

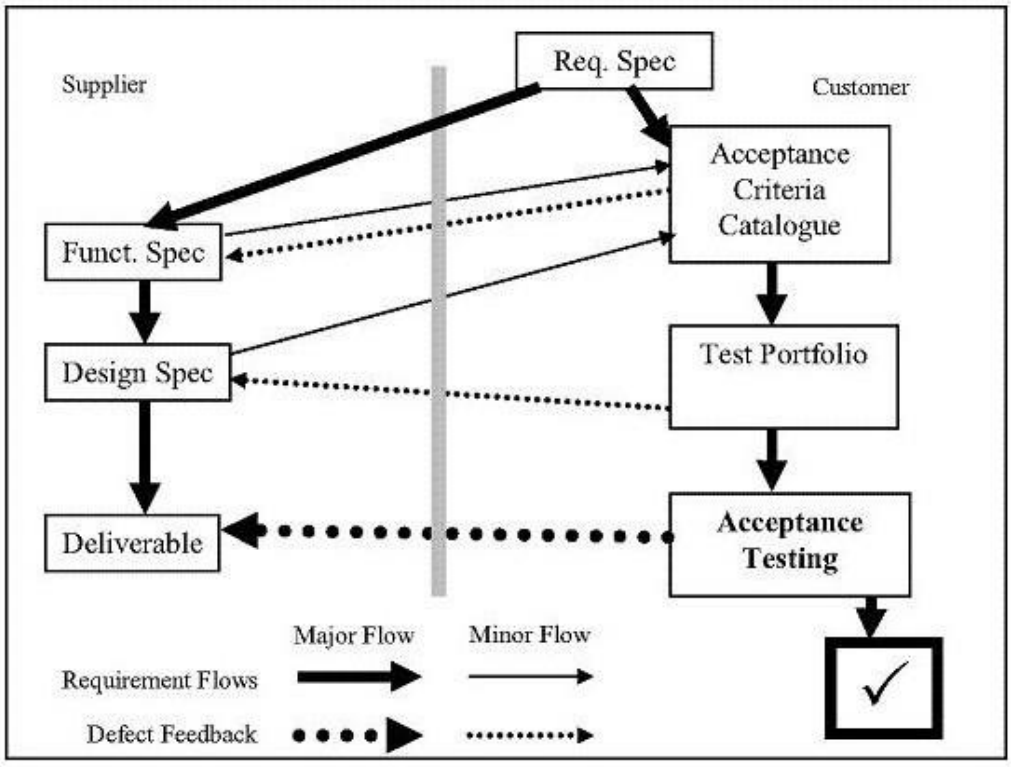



Figure 11. The components of user acceptance testing. (Windle 2010, 1).

Figure 11 above illustrates how the user acceptance testing is designed from the beginning. This figure gives an overview of what type of items that are sought after in designing the user acceptance testing. The supplier is on the left side of the figure and the supplier is responsible for functional and design specifications. The customer is on the right side of the figure. The requirement specifications are done from the customer's needs based on their business requirements. The specifications provide a great deal of information to the supplier's functional specification as well as the design specification.

The customer's requirements are grouped into an acceptance criteria catalogue or they can be grouped in some other method as well. The acceptance criteria catalogue contains all the user requirements. This will give them input in the test

portfolio. The test portfolio contains a grouped customer requirements in the same domain area, such as sales order handling.

This will eventually lead to the acceptance testing when the user acceptance plan is created. This is based on the user requirements and they have been sorted into smaller requirement groups. The acceptance testing produces qualified information to the supplier. With the help of this information the supplier can then base the deliverables or new testable versions to the customer. (Windle 2010, 5.)

It is important to understand what user acceptance testing is and is not. Here is the one definition of what user acceptance testing (UAT) is:

"UAT validates that users can perform their jobs using the new modified system and that requirements were given, documented, interpreted, designed and developed correctly" (Cimberman 2007, 8).

The UAT focuses on real business processes and suitability of the system based on real user scenarios. The UAT is done always before "going live". The user is a key. The users will accept or reject the system. The UAT is done by core business users and they will indicate if they are satisfied and the system meets their requirements (Kusiak 2010, 4). The UAT is the most important test in the software development field (Leung et al. 2007, 1).

According to Pyhäjärvi, acceptance tests are misnamed due to the fact you don't know when you're done when they pass; you know that you're not done when they fail. They should be rightfully named "rejection tests". (Pyhäjärvi 2012, 10.)

User acceptance testing is based on the understanding of the user's requirements. How the system works and how it is supposed to work. Customer should know what they want. A customer can only decide what expectation has to be met. (Windle 2010, 4.)

Acceptance testing is designed and performed by the customer. Acceptance testing could be done on behalf of the customer but this is not recommended.

Outsourcing the acceptance testing requires good documentation and very good communication between customer and outsourcing party. (Windle 2010, 24.)

User acceptance testing is always performed by the business. UAT includes business process testing, business scenario testing and high risk transactional testing. (Kohlman 2011, 13.) Testing always takes time and you should reserve enough time for planning and executing the user acceptance tests. In a citation below this is stated well.

"The inexcusable mistake in acceptance testing is to spend time and resources performing test and then fail, for whatever reason, to evaluate the results" (Windle 2010, 16).

### Importance of user acceptance testing

There are some important reasons as to why testing should be ranked as top priority at the start of the ERP implementation. For most businesses if IT stops, the business stops. Business goes live during the ERP implementation. It is a culmination of time, effort, resources and finance. Business users either accept or reject the system, so it is crucial to get business users involved in the UAT (Kusiak 2010, 1).

Implementation of new the ERP software is always critical for business. Therefore you have to concentrate on user acceptance testing and keep focused on the testing issues. A serious risk of the UAT failure could be a severe consequence from incomplete or rushed testing. This could put companies out of business (Kusiak, 2010, 4).

Poor testing can cause mission failures, significantly impact operational performance, reliability issues and raise costs to an unexpected level. On the other hand a good testing plan is a major project cost. You can make testing efficient. Make time for planning and organize testing properly. A good testing plan forces you to deal with problems as they arise. With a systematic testing plan the cost

of rework and fixes are much lower. (Kohlman 2011, 150.) The importance of the UAT can be narrowed in one citation:

"Testing is insurance, many have learnt the hard way" (Kohlman 2011, 180).

The importance of UAT is to look at the entire picture. Things will be complicated if there are two or more suppliers and they are responsible for part of the system. The UAT finds only bugs or defects – it does not define who is responsible for the cost of fixing them. (J Windle 2012, pers. comm. 14 May.)

The UAT can be seen as simulation of the "going live" phase. It tries to reduce the risks related to it. The UAT plan needs to be a written plan. Issues that have been found during the UAT will mean fewer issues that have to be solved after "going live". Testing needs to be planned and the defects need to be recorded when they have been found. (Roys 2008.)

The main objectives of UAT are to prove the selected software is suitable for its purpose. The customer can accept the work done by the ERP provider and the invoice can be paid to the provider of the system. (Munawar 2011.)

In addition, the objective is to catch bugs in the system during the warranty time. The provider is responsible for fixing bugs in the system and the business users' work with the new system can be productive from the beginning of production use. (Vuori 2010, 5-6.)

### V-model in user acceptance testing

The most used lifecycle model in software testing is the V - model. This model maps each development phase to a particular testing phase. This model was developed for the clear and distinct hands-offs between different phases of the project. It also informs who does what at each different testing phase. (Kusiak 2010, 5.)

Figure 12 illustrates that analysts and business users specify the requirement and need to take responsibility for the acceptance testing. It validates that the

user requirements have been met. Acceptance testing is concerned with what the system does. (Kusiak 2010, 5.)

Analysts, users and designers convert the requirements into logical high level design and they are responsible for specifying system testing. System testing validates that the system meets system requirements and design.

Software designers and architects specify how the system will be constructed in physical terms. Integration testing ensures that the units comprising the system integrate together into a successful working unit. Integration testing is also sometimes called as “link testing” or “end-to-end testing”, or “flow-through testing”. (Cimberman 2007, 8.) This phase is called low level design as well.

Programmers work from the specifications that are produced by the designers. Designers have done their test and now programmers will ensure that individual components meet program specifications. This testing validates that code works as designed (Windle 2010, 23).

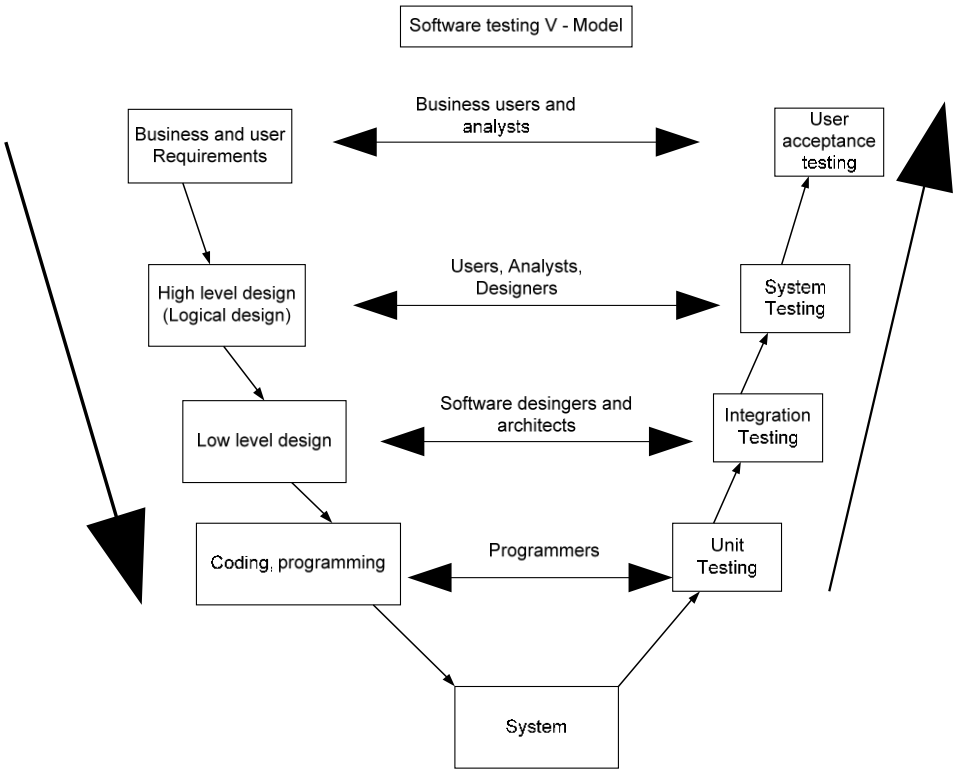



Figure 12. The V-model in software testing. (Black 2009, 501).

Software testing consists of unit testing, integration testing, system testing and user acceptance testing. Testing strategies in the V-model from unit to system testing are for the developers. Testing is a continuous process which carries on until the system is fit for their purpose or closely matching required goals. (Bordo 2010, slide 7.)

Exhaustive requirements and an IT system that meets perfection is impossible to do. This is why you should plan and design testing, perform it according to the plan and review the results you will learn from feedback. This process will continue until the result is adequate. (UAT process 2008, 4.) The PDCA process can be seen in figure 13 below.

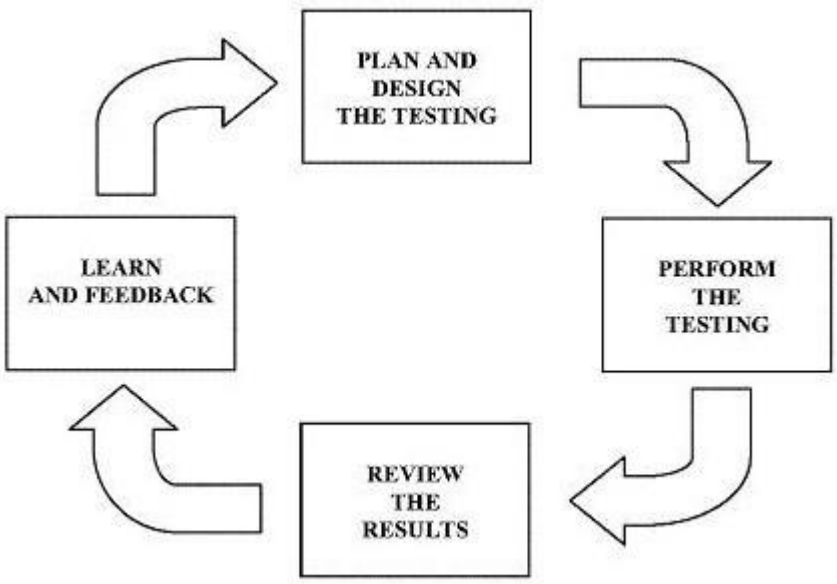



Figure 13. The iterative testing process (Windle 2010, 21)

This is an iterative approach and consists of requirements, analysis, design, implementation and testing phases. Each of these has to be completed before the next phase can start. This is the waterfall model in SDLC. (Kohlman 2011, 5-6.)

### What to test?

This question is important because it defines the objects that are needed for planning the UAT. For planning you need documentation, business requirements, functional requirements, business processes, workflows, transactions and the software. All these elements must be considered when placing the question “what to test?”. Also analyzing what to test is done based on these documents.

If the outcome is immaterial you should not expend time and effort. You should always ask “What if the requirements are not met?” also “What is the impact of not fixing it?” (Windle 2010, 6).

It is impossible to test every possible issue in each business system. The most important functionalities to be tested are business functions. Business users have to have the knowledge and understanding of their business processes. They will accept or reject the system if the answer to the question “what to test” fails. They have to live with those consequences. (Kusiak, 2010, 2.)

According to Black you can’t test it all. You should question yourself what must be tested. Selecting what to test is sometimes a very hard choice but under those circumstances you should consult a test stakeholder as well as the test team (Black 2009, 135).

### Why is it important to test?

The IT systems such as an ERP system, is a very complex software. Testing confirms that the requirements have been met. The primary reason for testing is to find bugs or defects. This is widely accepted wisdom in the testing industry (Kaner 2004, 4).

The reasons for testing are as follows:

- To ensure you are receiving what you asked for
- To protect you from accident or financial loss
- To identify defects with the aim of eliminating them
- To be able to negotiate when it comes to identified defects that can't be resolved
- To demonstrate the correct action after defects have been resolved
- To establish or verify the scope and limitations of what is being accepted
- To foster confidence in what is being accepted
- To become aware of limitations in the functionality. (Windle 2010, 5.)

You have to keep in mind that there are serious consequences of a defective system. The risks are related to loss of revenue or profit. These are obvious results of defects:

- The loss of business
- Damage to equipment because of a malfunction
- Bad publicity
- Training overhead
- Unnecessary clerical work
- Cost of corrections. (Windle 2010, 5.)

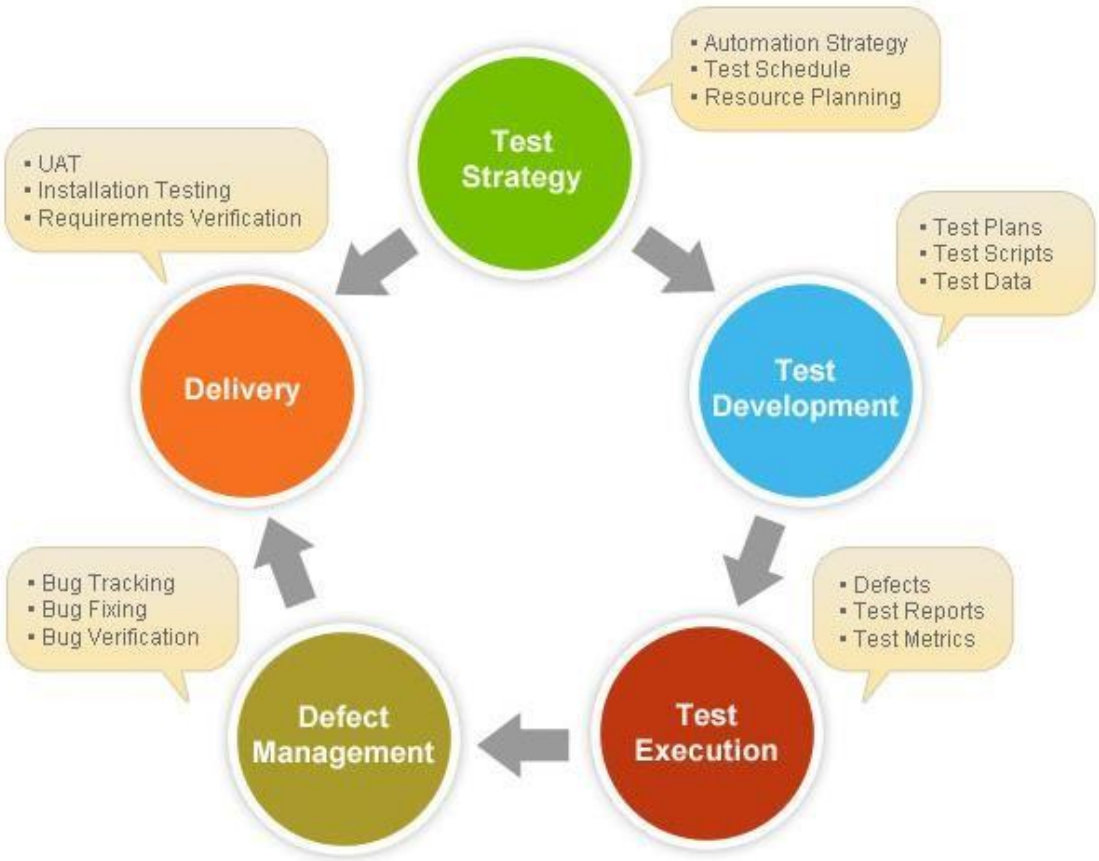



Figure 14. The steps in UAT. (Software testing help 2012).

Figure 14 above illustrates the steps in the UAT planning. The UAT actually begins with a test strategy. The UAT manager needs to decide what is the test strategy, schedule and resources. Test strategy also affects the delivery phase because the UAT is done just before delivery. The next step is to test development. It consists of test plans and test scripts (business scenario test cases). Test execution includes defects, reports, and metrics. The defect management consists of defect tracking and verification. Defect management takes care of testing tools such as the defect tracking system and screen capture tools. The delivery phase includes the UAT plan with verification and testing strategy. This step could be affected because a chosen strategy will dictate how the UAT is done in the delivery phase. (Software testing help 2012.)

### Acceptance criteria catalogue

A common nominator in the UAT designing phase is the acceptance criteria catalogue (Wikipedia 2013). The acceptance criteria are needed for sorting the relevant requirements into groups. The result of this procedure is an acceptance criteria catalogue.

The acceptance criteria catalogue is a comprehensive list made from the requirement specification document. It consists of all the requirements together with group identifiers. The requirements are grouped in to their own similar domain area.

When user acceptance testing is designed, the acceptance criteria catalogue will play an important role. Acceptance criteria are based on different types of documents such as requirement and functional specification. These are called sources for acceptance criteria catalogue. The functionality of the old ERP system is one possible source for acceptance criteria. The customer's task is to create an acceptance criteria catalogue from the requirement specification documentation. (Windle 2010, 34.)

Collecting criteria is a fast process when the sources for the acceptance criteria have been identified. The objective is to write one subsection for each requirement. (Windle 2010, 35.)

This acceptance criteria catalogue can be done, for example, with an Excel sheet. The acceptance criteria catalogue is used for sorting all requirements in the right order for the test script. The acceptance criteria catalogue consists of a group and requirement identifiers, criterions, test steps and description of criterions. This can be seen in appendix 2.

In my opinion without an acceptance criteria catalogue or similar document the design of the UAT is nearly impossible. There will be so many requirements in the beginning. This method will help to avoid the problem of not missing any requirements.

## Designing the test cases for UAT and testing techniques

Test case design is created with the business process owners in the designing phase after the acceptance criteria catalogue. Involving the business users in this phase is key for the UAT to be successful. The business users have inputs, while the UAT manager writes the test cases. (Rice 2009.) All test cases need to be documented and a good test case will find the defect. When the tests are successful they will reveal a defect (Kohlman 2011, 233).

The testing technique of how to input data to the system needs to be decided well before designing the test cases. There are positive and negative testing techniques. It is assumed that a system works as planned when valid data is input by the tester in positive testing technique. The basic idea of this technique is that the developer of the system has coded the functionalities to work successfully and the system does with it is supposed to do.

The negative testing technique is opposite of the positive. In the negative test invalid data is provided to the system. The negative testing technique tries to ensure that the system does not do anything it is not supposed to do. (Nyman 2002.)

Test cases can be divided into two categories. An individual test case checks a single requirement from the requirement specification. This is called specification based test case. The business scenario test case checks a complete process. It is called business scenario based test case. (Kaner 2003, 8.) An individual test case can be just one requirement in the acceptance criteria catalogue. In addition, fulfilling the user requirements in specifications does not guarantee a success (Mak 2009, 1). The tests need to be based on real-world examples. The business scenario test case tries to tackle this challenge. If the tests are not based on real world examples such as the sales order process, the point of the UAT is missed completely.

The requirement is verified by the individual test cases. Test business scenario's that are used for validating the company's business processes. Validations

by the business test cases are needed. This could be because requirements are not necessarily completed or business users that have participated in the workshops have not understood how to explain all requirements or they have assumed that it is obvious that the system behaves in a certain way. (Rice 2009.)

Test cases need entry and exit conditions. Entry conditions give an introduction to the test case. The tester knows what they are trying to achieve in the test case. In addition, exit conditions are expected results. They provide information to the tester what is the expected result. The tester needs to verify that entry and exit conditions of the test case are met. (Vyavhare 2012.) An example of the individual test case and its entry and exit conditions can be seen in appendix 3.

### Test execution

The test execution is a phase in the UAT which is done after the acceptance criteria catalogue and test cases have been created. The entry and exit conditions have test steps in place. In addition, the test team has agreed that the test plan meets the requirements of the business users.

The test environment, defect reporting system and testing tools such as screen capture software are set and running. Test execution begins with the “smoke test”. By doing the smoke test the testers try to ensure that the most important areas of the program will work. Smoke testing must pass before further testing commences. Smoke tests are typically simple tests such as the logging into the system with the user credentials is working. (Rouse 2006.)

The test execution can be started after the smoke test is passed successfully. Test cases, such as sales order handling. Various tests are run, results are being checked and possible defects are to be reported in the defect reporting system. Testing is usually performed till progress can’t proceed until supplier supplies new deliverable. (The software quality institute 2007, 1.)

Testing can be continued after the developers have released a new deliverable. At this point the testers have to ensure that all the defects in the previous release have been resolved in the latest release. Performing all tests that were not run before due to critical defects should run now. In addition, the testers need to do regression testing on the latest release. The regression testing confirms that the changed code has not negatively affected any features of the program. Regression testing ensures that the program or features behaves as it should. (Guru99 2012.)

When the tests are successfully completed, results are accepted by the testers and the tests are signed by the process owners. The test execution is an iterative process, it continues as long as defects are found and stops when the system is at an acceptable level by the process owners. (Windle 2010, 63.)

The UAT before “going live” is typically 1 – 2 days long. That particular period should be reserved only for testing. This phase should not be an error –fixing session. (Pyhäjärvi 2012, 68.)

### What is a defect?

A defect is an error, problem or incident that is found in the application which prevents the tester from executing a function which is required to happen in the test case or during the process. A defect can happen if the tester observes that the expected results in the test case does not match the requirement. (Reddy 2012.)

From the above description, defects can have many variations. Defects can be divided into five subcategories. Those categories are illustrated in figure 15 below.

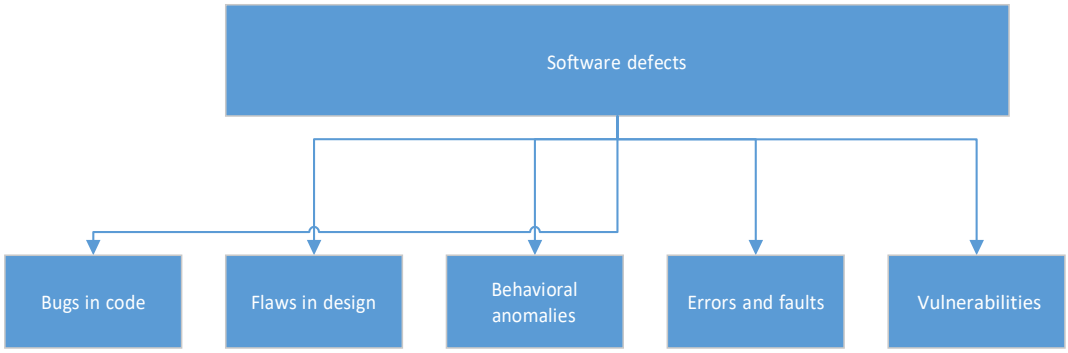



Figure 15. The software defects categories. (Paul 2011, 354).

Defects can be bugs in the code. It is executed in the system when it fails to do what it should. There can be flaws in the software design or human error which is causing the defect. In addition, errors, faults and vulnerabilities of the software are classified as defects. (Kohlman 2011, 246.)

The objective of defect reporting is that they are noticed by the system developers or system providers. They are reported and tracked by the defect tracking system. Reporting of the defects has to be very precise and detailed. For example “this is not working” is not a good defect report. (Paul 2011, 354.)

### Handling defects

The lifecycle of a defect starts when a new defect is found. The defect will be reported into the reporting system. Developers should resolve the defect and it will be closed after resolution. This can be seen in figure 16 below. This is a very simple model of the defect lifecycle. A defect includes a lot of information about the incident for defect tracking purposes.

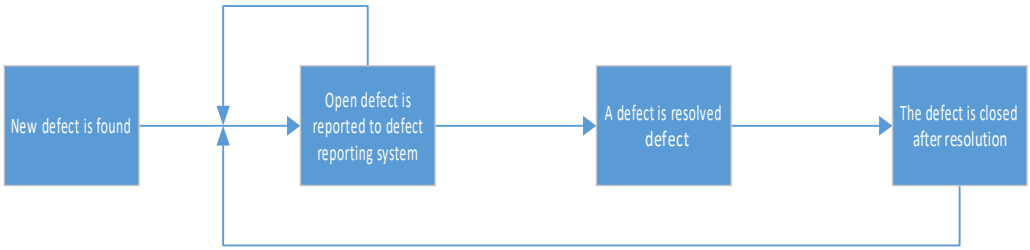



Figure 16. The lifecycle of a defect. (Windle 2010, 77).

Managing defects can happen with the defect tracking system. The following information is mandatory for the defect reporting to be an effective. A defect consists of following crucial information:

- Defect identifier
- Title of the defect
- Description of the defect
- Expected results
- Priority
- Severity classification
- Version
- Attachments.

The defect identifier is a unique number or an identifier. The defect can be tracked with this identifier easily from the defect tracking system. The title of the defect should be concise and clear. Description of the defect –field should have detailed and precise information about the defect itself, the steps that lead to the incident.

In the defect description the expected results should be mentioned, it is easier for the developer to understand what is supposed to happen. This ensures no guessing games are needed. (Paul 2011, 355.)

The priority and severity classification are needed for the developers. Placing priority can identify how urgent this defect is for the system under testing. Severity can tell if the defect is major which prevents the business use or is it minor which does not prevent use of the system. The version is needed for tracking the defect, for example the defect was reported in version 0.1.1 and it is fixed in version 0.2.1. (Kohlman 2011, 250.)

When the defects are reported to the tracking system. It is good advice to add a screenshot or other relevant documentation relating to the defect. This could be very helpful for the developers who are trying to fix the reported defect. (Paul 2011, 355.)

All the defects that have been transferred to the production environment could be very expensive for company. Also the company could lose its reputation if the defect is found in the production environment. Handlings of defects are the important part of the UAT process. The objective of testing is to use the system as it would be used in a real environment. (Pyhäjärvi 2012, 13.)

### The sign-off

The sign-off is done when the UAT is successfully completed. The sign-off is the final step in the UAT process. The sign-off documentation is the formal document for going live or not going live with the new ERP system. The document is signed by the person who is representing the ERP users and has the authority to do it. (Paul 2011, 354.)

The sign off is done for all the test cases and business test scenario test cases. The signed document is the verification that the system meets the user's requirements and is able to run company's daily business. The responsibility of the defects in the software does not end with this document. This document is to use to give permission to go or not go into the deployment phase. The sign off can be launched for example on following reasons

- The project deadline is reached

- The budget for testing is exceeded
- There is no point to continue testing, the system is tested enough to fit its purpose. (Windle 2010, 89.)

The decision to “going live” –phase is usually done in the steering group meetings with the company’s executives. The decision for “going live” must be based on facts. The signed test cases can help the executives to make the decision. They must have all the documentation available, for example the test results, known problems, outstanding issues and possible workarounds. The UAT manager representing the UAT team has the most influence and the most important issue is the business process testing results. (Cimberman 2007, 29.)

After sign off, the ERP system is ready for deployment. There are a lot of items to take care of such as, data conversion and transferring the ERP functionalities to the production environment. This is usually done on the weekends due to the fact that working with the current ERP system ends usually on the previous day.

The new ERP is in production use after the weekend. The conversion runs will take a lot of time and they should be rehearsed before the actual “going live” day is at hand. In addition, when the system is taken in use on the weekend, the possibility to resolve encountered problems is better without rushing. The ERP consultants should be on site after “go-live” for a couple of days to ensure that the new system is working accordingly.

# 4 USER ACCEPTANCE TESTING PLAN

Test documentation

Test documentation plays a key role in the UAT. It consists of several documents and issues that are related to the testing such as:

- Test strategy
- Testing responsibilities
- Test cases and approval process
- Test environment
- Entrance and exit criteria for the test cases
- Business scenarios to be tested
- Smoke tests
- Risks and mitigations
- Defect reporting
- Change management
- Testing personnel’s contact information
- UAT schedule and resources
- Training
- Traceability matrix
- UAT checklist
- Sign off –documentation. (Cimberman 2007, 34; IEEE829 1998, 1.)

Overall testing guidelines give the testers basic guidelines, such as how they can execute user acceptance testing. The document discusses test environment, defect reporting method, test cases, entry and exit criteria. It describes business processes which will be tested during the UAT.

The test strategy contains information regarding the testing approach. The approach can be manual testing, automated testing or agile methods such as ATDD. Usually this defines what types of testing tools are used during the testing and training needed for it. It also defines what metrics will be collected and what level.

The test strategy also defines how many configurations will be tested and it contains information regarding the hardware and software to name a few. These are based on the IEEE829 test plan. The IEEE829 is a standard in software testing and a good resource for making the test plan.

Testing responsibilities describes the test group members who are responsible for executing the testing. Test cases and approval process should be described in the test cases with markings such as passed or failed. Entry and exit criteria should be described into each test case as well. (IEEE829 1998, 5.)

Schedule and resources should be in the test plan. Schedule for testing and what resources it will require. Training is another element of the test plan. Training should be planned for the testing tools and schedule it in the test plan. (IEEE829 1998, 1.)

The user acceptance testing plan is done from various different documents such as project documentation, requirement specification documentation, software user guides as well as minutes of meetings and business process descriptions. The actual acceptance testing plan is done by using these documents as a source material for the plan. These documents are called references. They give input and support the test plan.

In addition, the acceptance criteria catalogue and test portfolio or similar documentation is needed for designing the user acceptance testing plan. Although

they are not included into the actual test plan they play a very important role when designing the user acceptance testing. In addition to the IEEE829 standard a description of the business scenario test cases, smoke test are needed, change management and sign-off documentation.

The needed documentation can vary from project to project but the basic elements of the test plan, should be in place. Test strategy, testing responsibilities, training, entrance and exit criteria, schedule, resources and risks should all be part of the overall plan.

### The test team

The test team is formed to ensure that the user acceptance testing meets its objectives. The test team usually includes an UAT manager and testers. The UAT manager's responsibility is to plan the UAT documentation and guide the overall testing process. Testers are in a key position when user acceptance testing is executed. Business process owners have the needed knowledge of how the ERP should behave in various different cases. Therefore it is very advisable to use end users as testers in user acceptance testing (Katara 2011, 111).

The structure of a test team is an important factor in the user acceptance testing. The test team should include both technical and domain experts such as business owners. When there are team members who have technical expertise and other members have business process expertise the test team is most efficient for completing the user acceptance testing. (Dustin 2002, 63.)

Most of the UAT literature authors talk about the ERP implementation project having their own test organization, but this is not necessarily true in the real world. SMEs do not usually have required resources to hire outsourced testers. Therefore ERP end users should be part of the test team. They have the needed expertise in the business process domain area. Organizations should use these in-house resources in the test team. They are experts in handling business processes on a daily basis. (Cimberman 2007, 62.)

Typical roles and responsibilities of the test team

The testers' main task is to find bugs or defects in the system and to prove that the ERP system works correctly. They focus on business process streams and ensure that they will work correctly while the system is under testing. It can be thought that the testers will ensure the quality of the ERP software. (Kaner 2004, 2.) The quality of testing depends on how the testing is done. Testing does not automatically guarantee the software quality (Leung et al. 1997, 137). Following common roles and responsibilities in ERP test teams are presented in table 3.

Table 3. Roles and responsibilities in the test team. (Dustin 2002, 66-70).

| Roles | Responsibilities | Skills |
|---|---|---|
| **UAT manager** | Head of the test team | Managing skills |
| | Communication with the ERP provider | Communication skills |
| | Designing the UAT plan | Understanding of test process strategies |
| | Writing test cases | Understanding of test process strategies |
| | Outlining the test schedule | Project management skills |
| | Defining the test strategy | Understanding of test process strategies |
| | Selecting the test tools and introducing them to the testers | Familiar with test environment and different testing tools |

(to be continued)

Table 3 (continues)

| Roles | Responsibilities | Skills |
|---|---|---|
| **UAT manager** | Planning the test environment | Project management skills |
| | Acquiring the needed software and hardware | Familiar with hardware and software |
| | Coordinating the UAT team meetings | Project management skills |
| | Reporting the status of the project to the test team and executives | |
| | Quality Assurance | |
| **Testers / End users** | Planning business scenario test cases with the UAT manager | Business process knowledge, subject-matter expertise |
| | Executing the test plan | Attention to the details, ability to see “the bigger picture” |
| | Reporting defects to the defect reporting system | Documentation skills |
| | Communication between with the UAT manager, ERP vendor and other team members | Teamwork and communication skills |

Roles and responsibilities of the test team can include many various tasks. What kind of test team is required depends on the project. To complete objectives of the project, the most important thing is to form the test team which has required skills and expertise. These factors are required in building an effective test team. (Acharya & Iyer 2012, 1.)

### The test environment

The test environment should be dedicated and isolated from the development and production environments. The test environment should be a complete replica of the production environment. (ISO27001 2005, 18.) For the UAT a dedicated test environment is preferred. This environment should not be used for any other purpose at all.

The dedicated test environment provides a stable platform for testers to execute the testing process. Testing can be done without fear of sudden version updates or interference by a 3rd party, such as developers. A test environment protects and ensures that no testing is done in the production environment. (Windle 2010, 155.)

There are a lot of resources required in building this setup, but today with virtualization resource issues are reduced. Keeping in mind a testing environment does not guarantee a working ERP in the production environment.

A test environment provides a platform where all necessary testing must be done before the UAT. The developers and the testers should have easy access to this environment. Together with the deliverables which are installed into this environment on a regular basis.

The deliverables are based on versions. Information on the new deliverables must be sent beforehand. For example send it by email to the UAT team. The test environment must be stable, working and the testers need to know the current version for testing test cases. (Pyhäjärvi 2012, 155.)

Companies which run the UAT can have at least 3 different environments such as development, production and the test environment. The challenge is transferring all the settings made by a user, developer or by the system to another environment. This requires a detailed migration plan.

### Risks in UAT

Understanding the risks in the UAT is an essential part of the UAT design process. The risks need to be identified and the objective is to mitigate the risks that are related to the UAT. These are the most common risks:

- Misinterpretation of requirements
- Hostility between users and the provider
- Lack of user dedication
- Lack of resources
- Tight time schedule of the UAT. (Allgood 2008, 10-11.)

Misinterpretation of the requirement is a risk that could occur at the very beginning of the UAT. The UAT is the based on the requirement specification documentation. The risk is that the requirements are too broad and vague. Hostility between users and the provider can happen if there is a misunderstanding of the written requirements. Lack of user dedication could be one possible risk in the UAT.

The UAT could not have enough allocated resources or assigned resources. Sometimes users are not interested in taking responsibility of the UAT. This usually happens when users are transferred to another project before the first is even finished. In addition, the schedule of the UAT is at risk as well. (Allgood 2008, 10-11.)

Some product risks could be caused by a version change in the middle of the project or a data conversion took more time than anticipated. The risk analysis is needed for the UAT project. (Pyhäjärvi 2012, 190.)

There are a lot of risks that can relate to the UAT project which come from outside, of UAT project, such as a subproject during the ERP implementation. Those risks are usually based on time scheduling. Other risks are complexity of the product, business risks or cost of faults. (Allgood 2008, 10.)

To mitigate the above risks, evaluation needs to be done by a risk analysis. Figure 17 below illustrates the steps of risk analysis. All risks need to be identified and after that an assessment of the risk finally should be done. Magnitude, likelihood and the risk level are all considered. Lastly, you should be able to minimize the risks accordingly.

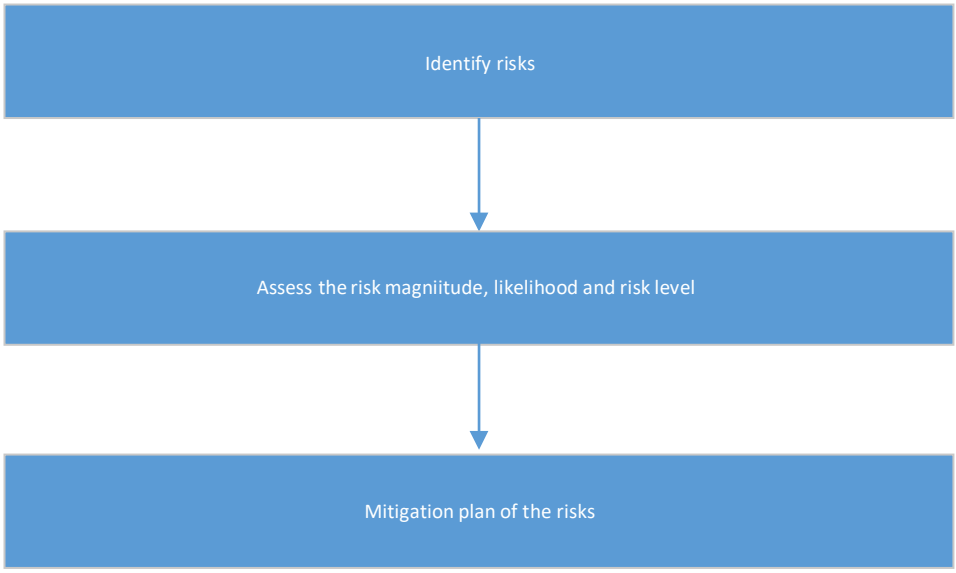



Figure 17. The risk identification process.

The identified risks are put into risk level matrix. Risk control recommendations are needed. The risks can be mitigated to an acceptable level, eliminate the risk or just accept a risk as it is.

# 5 INTRODUCING THE ERP PROJECT IN SATEL

Why the project is conducted?

SATEL has been using many versions of Digia Enterprise ERP system in the past eleven years. It started from 1997. SATEL was using a software called Työkalupakki and the main challenge was that production processes were not documented at all. Production processes were executed from the production manager's memory. Problems started to occur when the company hired a purchasing manager. The purchasing manager could not purchase components because there was no centralized ERP system available at that time. The objective was to switch Työkalupakki –software to the Powered 2.2, later known to be as Sentera Enterprise and Digia Enterprise. The migration from Työkalupakki to Powered 2.2 was not an entirely a success, the budget for the ERP implementation exceeded about 200.000 euros. This was made clear from our CEO who was in the management group of that project.

The reason for this was due to the fact there was no specific business requirements and most of the functionalities was copied from an ERP system. This other ERP system was tailor-made for another company that operated in the same manufacturing field. Another reason for not sticking to the budget was that there was no systematically planned user acceptance testing. This time the ERP project differs from earlier versions. We are changing to an ERP system called Microsoft Dynamics 2012 AX.

This time user acceptance testing plan for testing has to been made. There are major risks related to changing the entire system. Risks that are related to the business and reputation of the company.

Our current ERP is from 2006. It is a quite a long time for one ERP system and the company's business has changed, added new products with modular structure and advanced. The current ERP does not support functionalities that the company's stakeholders require. This client software does not support Windows

7 operating systems and we can't start a development project with the current web shop. The current ERP has become an obstacle in our business today. These are the main reasons for the ERP implementation and these are project to only name a few.

SATEL has started the ERP implementation project in the beginning of 2012. Microsoft Dynamics AX 2012 ERP was chosen after a comparison of other competitive ERP products such as SAP, Digia Enterprise and Microsoft Dynamics NAV. Finally Microsoft Dynamics AX 2012 was chosen for SATEL's new ERP system. This was based on the results from the fit-gap analysis and proof of concept.

### Short introduction of Microsoft Dynamics AX 2012

Microsoft Dynamics AX 2012 is the chosen ERP software for SATEL. It is the latest version at the moment. Microsoft entered the ERP markets by acquiring Damgaard in 2004. Microsoft Dynamics AX 2012 is the ideal ERP software for mid-sized companies. It meets the requirements of multinational companies as well. (Luszczak 2012, 1.)

Microsoft Dynamics AX 2012 is an adaptable business solution that supports industry specific and operational business processes. It meets the complex requirement needed in manufacturing and in distribution. (Microsoft 2013; Tvision 2013.) It can be installed in a very simple environment or in a complex multinational environment. It is integrated deeply with Microsoft technologies such as Microsoft SQL server, Microsoft Sharepoint, BizTalk Server, and Microsoft Office. (Luszczak 2012, 2.)

Microsoft Dynamics AX consists of following functional components:

- Sales and marketing
- Supply chain management
- Service management

- Production
- Financial management such as accounts receivable, accounts payable and general ledger etc.
- Bank management
- Business Intelligence and reporting
- CRM
- HRM
- Projects (Luszcak 2012, 2; Wikipedia 2013)

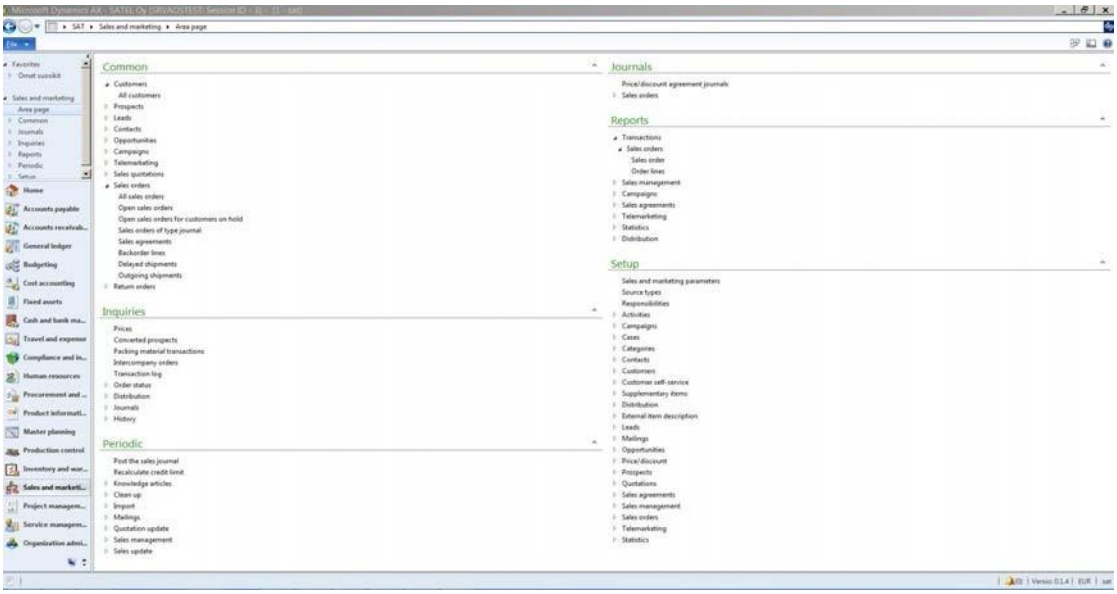



Figure 18. The main page of Microsoft Dynamics AX 2012 ERP –software.

Payroll management is a vertical solution because each country has their own legislation for payroll. In addition, you purchase several vertical solutions for the Dynamics AX such as web shop, electronic invoicing etc. Vertical solution is a separate software that can be integrated into Dynamics AX. Figure 18 above illustrates the main window of Microsoft Dynamics AX 2012, Functional areas such as sales and marketing, production etc. can be

found on the left pane. On the right pane are the main functions of the modules.

### Objectives and challenges of the ERP project in SATEL

There are several objectives for the ERP project. SATEL wants to replace and optimize manual routines done for example by Excel sheets. In addition business units require expense, income and budget monitoring. This is not possible at the moment in the current ERP system. Additionally, the new ERP system will handle the parent company and its subsidiary accounting in the same ERP system.

SATEL has had challenges to maintain access rights in our current ERP system. They have been very difficult to change if some special user rights have been required by the business owners. For example if the business owner wanted to have “read only” user rights, to have those rights required several hours of work for the IT person because user rights were not based on user roles. In Microsoft Dynamics AX 2012 all the user rights are role based and the system has a built-in role for the read-only users. This kind of feature is very popular amongst top level management in SATEL.

SATEL will have new layers to products and label management. SATEL’s new radio modems are based on modular product structure and it is not possible to manage BOM(s) on these products with the current ERP system. In addition, the tailor-made software changes will be reduced significantly in Microsoft Dynamics 2012 AX. This will be done because it is very difficult to maintain and update heavily tailored ERP system as well as being costly.

On the other hand tailored features are sometimes absolutely necessary because it is essential to our business process. The objective is to minimize the needed tailored features. In the current ERP system (Digia Enterprise) there are about dozens tailored features. In Microsoft Dynamics AX 2012 there will be only 8 – 12 tailored features. The significant difference between Microsoft Dynamics AX 2012 and Digia Enterprise is that in the Microsoft’s ERP there is no

maintenance costs of the tailor-made software changes. This brings significant cost reduction in comparison.

One of the challenges that SATEL has with the current system is limited user licenses. The current system can have 10 concurrent users. This has been very challenging to the company lately. Concurrent users have been exceeded continuously. Some of the ERP users are unable to login into the system periodically. Therefore the company's new system will have more concurrent user licenses, up to 36 concurrent users and the users will be able to login in simultaneously.

SATEL's previous ERP operates on Progress 10.1A database. This database is the engine to an ERP system. The company has many databases related to operations such as testing, production and label printing. These databases are tailor-made for the company and the challenge is that running different conversion runs from the Progress database to Microsoft SQL databases is difficult and costly.

The company would like to have one uniformed platform for all databases. In this case Microsoft SQL Server 2012 simplifies the database management and is needed under one database engine. The new ERP enables better integration and compatibility to other Microsoft based programs such as Microsoft Office, Microsoft Lync and Microsoft Outlook. This will speed up daily work routines because the different programs are from one software provider. The deep integration in Office products is one benefit that we can achieve by Microsoft Dynamics AX 2012. It has Microsoft Office add-ins which enables the copy / paste feature to work with any Windows application. (Luszczak 2012, 45.)

Finally the collaboration features of Microsoft's ERP are the enterprise portal and AIF. Enterprise portal grants direct access to the ERP through an Internet browser and access to the data can be limited by the user role. An enterprise portal can be deployed to internal users, external customers and vendors. (Luszczak 2012, 3.)

Application Integration Framework (AIF) supports automatic data exchange with business applications inside and outside the company. You can send and receive invoices, price lists, packing slips in XML format. (Luszczak 2012, 3.)

SATEL needs a new online store and capabilities of Dynamics' enterprise portal have been thought-out carefully. The web shop that is used today is obsolete and does not support online business at all. On the contrary, it prevents it. We have configurable products and we are not able to add new products to our web shop internally. We need to order coding work from Digia. We need to eliminate this process.

### The scope of the project

The ERP project includes CRM, E-business portal and the calculation of salaries module. In the future we are planning to start an electronic invoicing project. The company will renew our document management system as well as business intelligence software from IBM Cognos to Microsoft's BI.

During the ERP project we are only focusing on renewing the ERP and the salary module will be a vertical solution to Microsoft Dynamics AX 2012. The calculation of salaries –module is not included in Microsoft's ERP because it has many rules and regulations based on each country's own legislation. On the other hand it is possible to extend Microsoft's ERP functionalities by various different vertical solutions. Customers are required to buy a vertical solution for the calculation of salaries for this reason.

Considering this SATEL has bought the HRM module from Mepco Oy. This HRM module is in the scope of the project as well as E-business site and the CRM.

With the HRM the company receives human resources management and calculation of the salaries module integrate with Microsoft Dynamics AX 2012. In addition the company keeps its training register up-to-date in our HRM module

and it can keep up-to-date development discussions, first aid trainings and ISO9001 and ISO14001 standards.

The ERP implementation project contract was made with Nortal Oy in 2012. The choice was made through a selection process. We met with three possible ERP providers. Nortal Oy had the best price/quality ratio. They offered the ERP implementation project with a fixed fee. The company is starting to do business with the ERP implementations projects. Those were highly important factors for choosing Nortal Oy.

### The timetable and the value of the project

Preparations for the ERP project started already in 2010. By virtualizing the entire IT infrastructure on VMware virtualization platform and after that it had Windows 7 / Office 2010 implementation as well. Afterwards they changed the email system from Lotus Notes to Microsoft Exchange / Outlook system and added the Lync instant message / video conference system. All these previous projects were done because of the need for a new ERP system. The company required deep integration with different business software. Microsoft was an obvious choice for the company.

The ERP project with Microsoft Dynamics AX 2012 was started in March 2011. First the fit-gap analysis project was run. It took about one year to complete. The company needed to make sure that the chosen ERP system was the right choice for it. The company was making sure that the system could handle the requirements set by business processes and stakeholders. This is called diagnostic phase in the Microsoft's Sure Step methodology and it is the first phase before the ERP project begins.

The ERP project started in March 2012 and it has taken the entire year. The project has strictly followed Microsoft's Sure Step methodology presented earlier in chapter 2. The next phases are analysis and design that require a lot of user participation. Many workshops are needed during the project. It is very

time consuming from the customer's point of view as well as from the ERP provider's.

The ERP project should be completed during the second quarter of 2013. The company is aiming at May-June 2013. Smoke testing started in February 2013. The objective is to run the user acceptance testing during March - May 2013.

The value of the ERP project is ca. 500 000 euros and it includes the diagnostic project, tailor-made software changes, conversion from the old ERP to new ERP, proof of concept, all necessary licenses and hardware as well as software.

After the Microsoft Dynamics AX 2012 ERP is implemented. SATEL will launch the web shop project which has deep integration with the Dynamics AX 2012 ERP software. In addition, SATEL will start a Microsoft CRM project. The deep integration of the ERP, CRM and Enterprise Portal (online store) were kept in mind when the ERP implementation project was started. These decisions are based on the IT strategy of the company written in 2010.

# 6 DESIGNING AND IMPLEMENTING THE USER ACCEPTANCE TESTING PLAN IN SATEL OY

## The testing team in SATEL

The testing team was formed in November 2012. The testing team has 14 members and they are the main users of the ERP system. The main purpose of the testing team is to take care of testing of Microsoft Dynamics AX 2012 ERP. The team members were selected carefully based on their expertise in their business processes. Their expertise and knowledge is crucial for a successful and effective testing. They have a key role in executing various business processes in SATEL on daily basis. They are the best choice for executing the user acceptance testing for the company.

There are several objectives for the testing team. The information related to the ERP project can be distributed to the members of the team easily for example status updates and what will happen next during the project. The team can discuss issues concerning testing and deal with them.

Secondly, the business test case scenario design is done with the test team. The test team members have very detailed knowledge and expertise of the company's business processes. It is very important to use the team members' valuable knowledge and expertise in designing the business test cases.

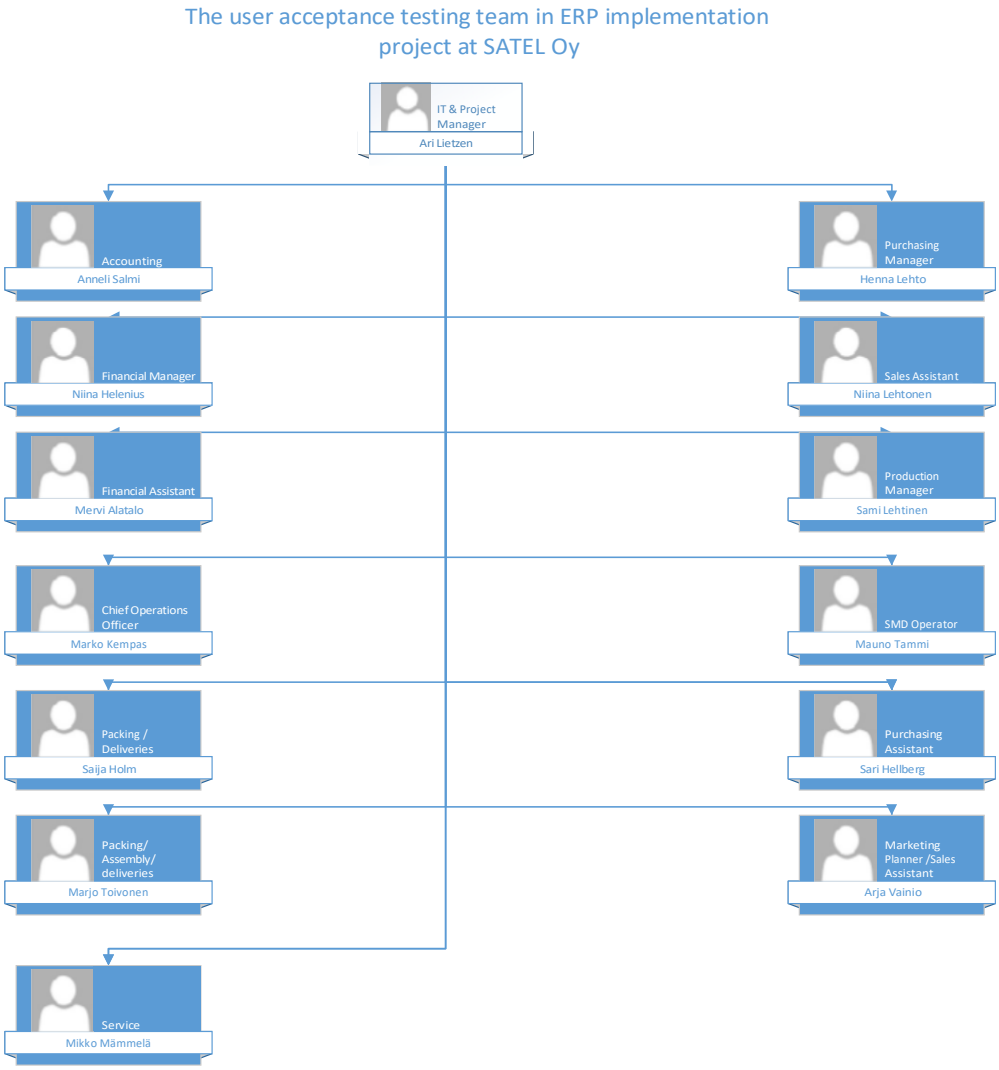



Figure 19. SATEL Oy's user acceptance testing team in ERP implementation.

In addition, the team must be trained to use the test plan and defect reporting system in order to complete the testing successfully. Training must be done before the testing starts. It is important the team can use the testing tools efficiently. In figure 19 above the team members and their roles are presented.

The test team members have required business process knowledge and expertise in using the ERP software. This is a vital requirement in designing the UAT. The detailed description of the test team roles can be seen in below table 4.

Table 4. Description of the test team roles.

| The roles | Short description |
|---|---|
| **Accounting** | responsible for accounting in SATEL and for the payment traffic |
| **COO** | responsible for R&D department, member of the management group and responsible for HRM related issues |
| **Financial manager** | responsible for calculation of salaries, manager of the Finance department and taking care of consolidated financial statements |
| **Financial assistant** | responsible for invoicing, intrastat and deliveries to the non EU-countries |
| **IT manager** | takes care of the company's it infrastructure, it support, maintaining the company's it environment, project manager for the ERP implementation and test manager for the UAT |
| **Marketing planner** | takes care of marketing related issues at the company, maintaining company's website, subsidiary to Niina Lehtonen |

(to be continued)

Table 4 (continues)

| The roles | Short description |
|---|---|
| **Production manager** | responsible for production and production planning |
| **Purchasing manager** | responsible for purchase department, inventories material planning, warehouses, purchasing, contacting with the vendors |
| **Purchasing assistant** | responsible to make purchase transactions with the vendors |
| **Sales assistant** | responsible for the sales order handling, service coordinator |
| **Service** | providing service for the products |
| **SMD Operator** | responsible for the planning of the SMD production and responsible for assembly machines |
| **Packing / Delivery /Assembly** | Packing the products, printing work item lists and making deliveries within EU (bill of freight) |

Responsibilities are divided by the business processes among the test team. The business process testing is the most important part of the UAT. It ensures that company is able to do their daily business with the new system and the objective is to cover many relevant business transactions. This method also ensures that all business processes will be tested by assigning the responsible persons for testing the processes. The testing team responsibilities are described by the processes in table 5 below.

Table 5. The testing responsibilities by business processes.

| The process | The responsibility |
|---|---|
| **An accounting /invoicing process** | Financial manager Niina Helenius, Finance Anneli Salmi and Financial assistant Mervi Alatalo |
| **A production process** | Production manager Sami Lehtinen, Saija Holm and Marjo Toivonen for the packing / deliveries |
| **Projects** | COO Marko Kempas |
| **A production planning process** | Production manager Sami Lehtinen |
| **A purchasing process and IWM processes** | Purchasing manager Henna Lehto, Purchasing assistant Sari Hellberg |
| **A sales order process** | Sales assistant Niina Lehtonen, Arja Vainio |
| **A service process** | Sales assistant Niina Lehtonen, Service Mikko Mämmelä |
| **An SMD process** | SMD operator Mauno Tammi |

Lastly, the test team is responsible with the UAT manager that all individual test cases, the business test cases are conducted and the objectives are achieved. The ultimate goal is to carry out successful UAT.

The UAT manager is responsible for coordinating the test team meetings, designing the test plan and training of the test tools. This is done with co-operation with the test team members. The test team meetings are held once a week or

more often if needed. It is very important to meet on a regular basis. The information is updated constantly and this pace with the meetings will guarantee that it will be distributed within the test team.

## The Introduction of sales order business process in SATEL

SATEL has various processes to conduct in its daily business with the customer. They are very common processes known within the ERP system as well. These processes are in use in other organizations. In order to conduct business the company needs all relevant processes to accomplish the business transactions within the ERP system. Figure 20 below illustrates all key business processes in SATEL.

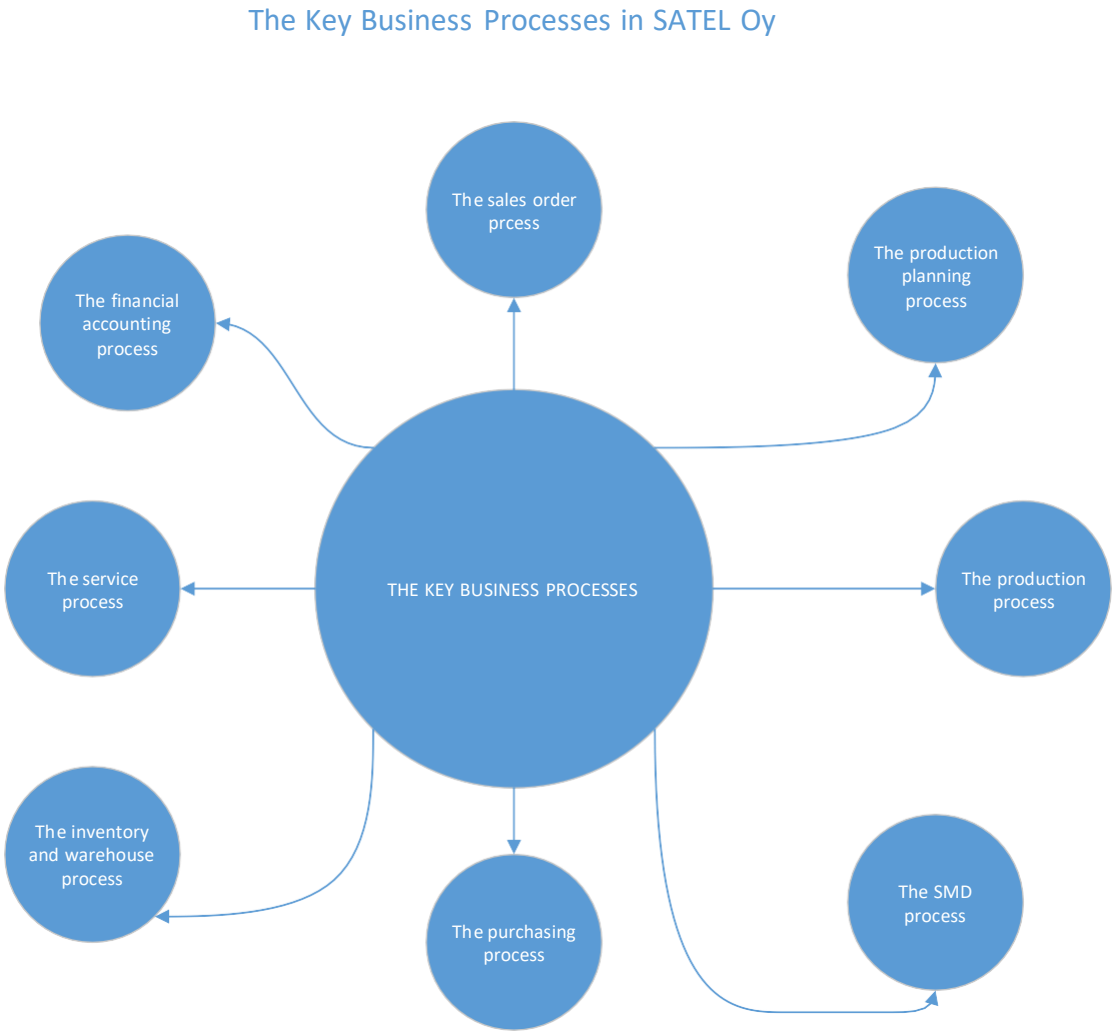



Figure 20. The key business processes in SATEL Oy.

In this chapter we will concentrate on the sales order process in the company. After the process is introduced, the test plan is presented and how it works with the sales order process. Figure 21 below illustrates the sales order process. The process starts from the customer, who will send a purchase order to the company. The purchase order is received usually by an email, a fax or from the web shop. We manufacture products only according to the customer's purchase order. We do not manufacture products in to our stock, because our product is configurable. It is tailored with unique settings such as radio frequency and channel spacing. To us the received order from the customer is called a sales order.

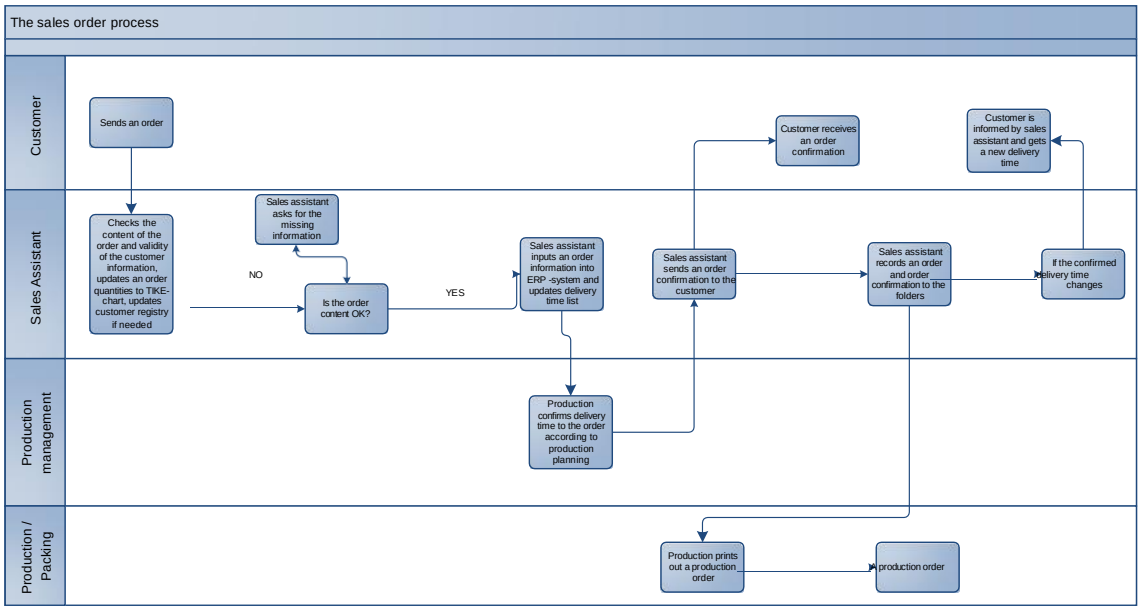



Figure 21. The sales order process in SATEL.

A sales assistant checks the content of the order and validity of the customer information will be updated as well if needed. There could be some incorrect information for example with the radio modems frequencies or channel spacing. After this the order quantities are updated to the TIKE –chart. The Information in the TIKE –chart (development of the sales orders) will tell how many radio modems and which models have been ordered on a weekly basis.

This information is used in a weekly chart which informs the stakeholders of how many orders came during the week, what is the invoicing amount of that week and comparison to the previous years. Lastly, the customer registry will be updated if needed. A sales assistant verifies that the order content is OK if it is not; a sales assistant asks the missing or incorrect information from the customer.

Picture 2. Testing the sales order process. (Sales assistant Mrs. Niina Lehtonen from SATEL Oy and Senior solutions specialist Mr. Pertti Anttila from Nortal Oy).

In picture 2 above Sales assistant Niina Lehtonen is inputting the customer's purchase order into the ERP system. She is carrying out the sales order process.

The next step is that a sales assistant inputs an order into the ERP system and updates the delivery time list. After this step has completed, the following happens in the production management:

A production manager confirms the delivery time to the sales order according to the production planning process. The products are manufactured according to the customer's purchase order. The products are not manufactured from the stock at all. After this step is completed, the process continues with a sales assistant. A sales assistant sends an order confirmation to the customer via email. The customer receives the order confirmation from a sales assistant.

After the sales assistant has sent the order confirmation to the customer, she records the order and the order confirmation to the specific folders which holds all the orders and the order confirmations. If the delivery time changes, a sales assistant informs the customer and the customer receives a new delivery time. Now the sales order process has been described completely.

Then the process continues to the production process. Finally the production personnel print out all production orders. With these orders production can assemble the needed products.

The previous description was the sales order process generally in SATEL. There are other processes that are tightly integrated with the sales order process. The processes are illustrated in figure 20.

Next we take a look in detail, the fulfillment process within the ERP system. This is a very important step because it is highly related to the UAT. It discusses how the individual test cases are mapped in the process.

This process description proceeds from a customer's purchase order to printing out the invoices to the customer. The complexity of the process testing is illustrated by this fulfillment process. This process includes several organizational levels and several integrated processes.

This is an overview what happens at the system level. Figure 22 below illustrates the detailed process interaction. Initially a customer sends a purchase order. Then a new sales order and the sales item lines are created. After this step the purchase order type is changed into a sales order. This step generates an input to the production planning process. From changing the order type it will create a production order.

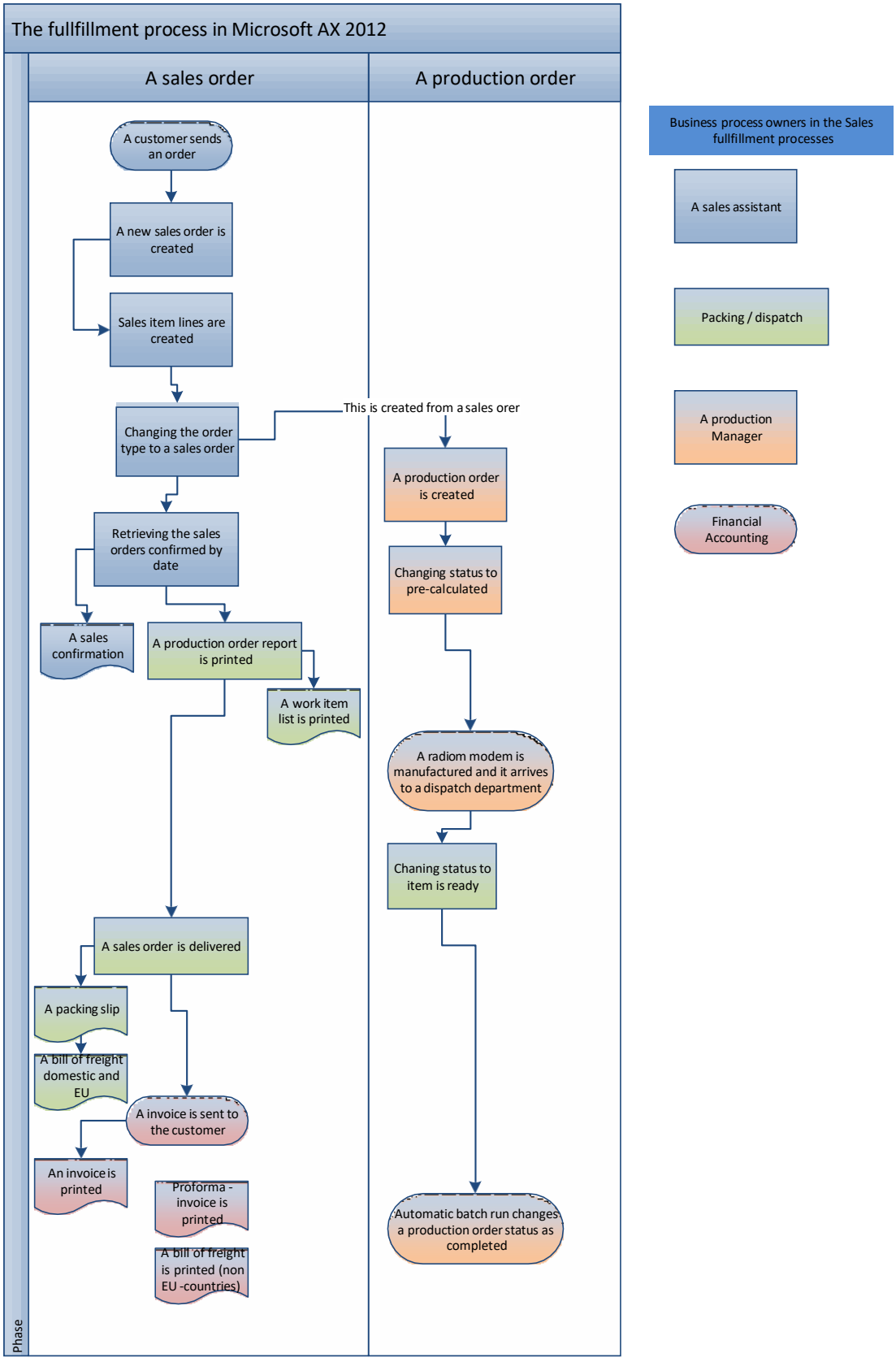



Figure 22. The fulfillment process in the Microsoft Dynamics AX 2012 ERP.

A production manager must pre-calculate the order status. This procedure creates the needed sub production orders. Keep in mind that these processes happen simultaneously.

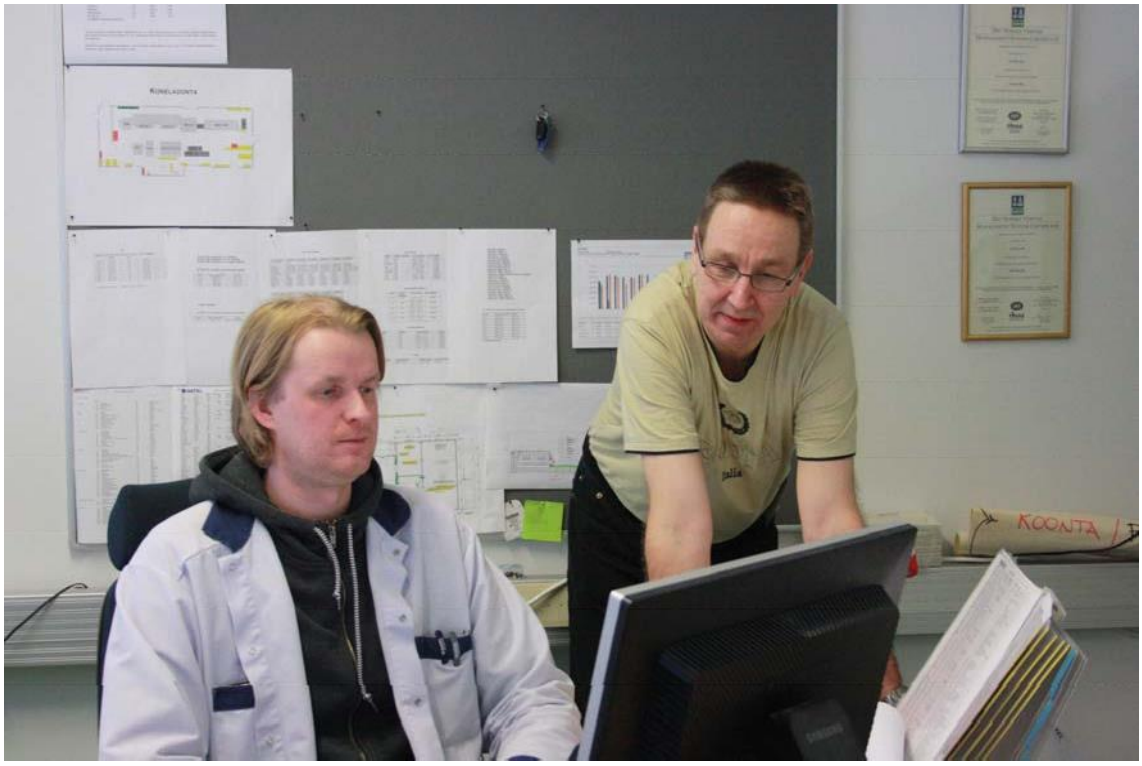
The sales order process continues after changing the order status to a sales order. The sales assistant retrieves the sales orders that the production manager has confirmed with dates from the production. After, the sales assistant can print out an order confirmation to the customer.

Now the packing personnel print out a production order report and a work item list for the assembly personnel in the production department. A work item list is printed weekly on Fridays. It guides working schedules in the production department. For example what type of radio modems will be manufactured to the customers according to their purchase orders.

The radio modems are manufactured by the production personnel. The products arrive to the dispatch department. Dispatch –department carries out packing of the products and reports the items as ready to be delivered to the customer. In this phase serial numbers of the products are created and the production is closed.

A packing personnel delivers the sales order by printing a packing slip and by creating bill of freight (domestic and EU). Next financial accounting prints and sends an invoice to the customer by paper or email. Financial accounting is responsible for creating and printing bill of freight (Non-EU countries) and a pro forma invoice for customs. Finally, the automatic batch run changes the production order status as completed.

This was a detailed demonstration of how the fulfillment process works but keep in mind that there are several processes that have interactions with other processes such as, procurement process, inventory, warehouse processes and financial accounting to name a few.




Picture 3. Testing the production planning process. (Production manager Mr. Sami Lehtinen from SATEL Oy and Senior solutions specialist Mr. Pertti Anttila from Nortal Oy).

In picture 3 Production manager Sami Lehtinen is testing the production planning process with Senior solutions specialist Pertti Anttila. This process is performed after the sales department has received the customer's purchase order and inputted the information into the ERP system

### The test plan

The purpose of the test plan in SATEL is to provide the test team guidelines explain how testing is done systematically. The test plan of the Microsoft Dynamics AX 2012 ERP implementation covers various subjects that are very relevant to the testing team.

The document gives detailed descriptions about testing, testing environment, defect reporting change control management, testing team responsibilities, roles, test tools, testing schedule, risk assessment and analysis. In addition, it

contains detailed information of the testing process, detailed description of the company's business processes and finally the test cases with entry and exit criteria. The test plan is based on IEEE 829 standard in software testing.

### How the plan was designed in SATEL?

The UAT process started designing the UAT strategy and approach by the UAT manager. The UAT manager chose the test strategy and testing approach. The choice was made between different methods including agile methods, automated testing and manual testing.

We decided to go with the manual testing based on the following issues this was a one-time project, automated testing and agile methods are usually used by the developers. The business processes are usually complicated in the company and can be based on executives' feelings. Overall benefits of the automated testing are minor. This was stated by Vuori (2010, slide 46).

In these cases manual testing approach is better. The agile methods require a lot of resources and communication. Usually the customer is with developers on a daily basis. The benefits of agile testing methods are not agreed by everyone. It is not necessarily faster, better, or not even cheaper than the traditional approach with manual testing. (Brown 2012.)

In addition, The ERP implementation project has a deadline. The agile methods are not necessarily dependent on a time schedules or deadlines. A customer usually can be flexible with the time schedule. For these reasons we chose to do the UAT with manual testing approach. (Vuori 2010, 46.)

In this first step the UAT manager also decides how to write test cases and the grouping of the requirements into similar domain area is done. In this step an acceptance criteria catalogue and test portfolio were created. The second step is taking the process owners into the designing phase. Together with them, test steps were given for each of the requirements. This was done for each pro-

cesses and requirements. The criteria catalogue was designed in a way that it was presented in chapter 3.4.

Next phase is designing the individual test cases against the user requirement specification. We had almost 400 user requirements. The test cases were written by the UAT manager. It took over three months to write all test cases. This was based on the theory found in (Kohlman 2011, 233). In addition, James Windle warned me of the heavy workload of this phase. I completely underestimated it.

After the individual test cases were designed and completed. The additional meetings with the business process owners were kept. In these meetings the individual test cases were mapped to each of the process, one by one. They tried to achieve the logical order of the test cases in this process with the business process owners.

Next, the UAT team designed the business scenario test cases with the UAT manager. In this phase the knowledge and expertise of the team is important to achieve quality, detailed and versatile business test case scenarios. These business test case scenarios are approved by the business process owners, although they can be the same individuals than in the UAT team. Designing the business scenario test cases were based on the theory found in (Kaner 2003, 8).

An execution of the acceptance testing is carried out next. When the defects have been found, they are reported to defect reporting system. Then the new deliverable is waited for and the testing starts again. This is also called regression testing.

This phase includes smoke test runs, many different testing runs as well. All the defects are reported to the BT.net. This testing tool is discussed in the next chapter. Finally, all critical defects to business are sorted out. Then the ERP system can be transferred into go-live phase. There might be issues such as historical information in the databases. That are planned to be sorted out after go-live but they are not critical to daily business.

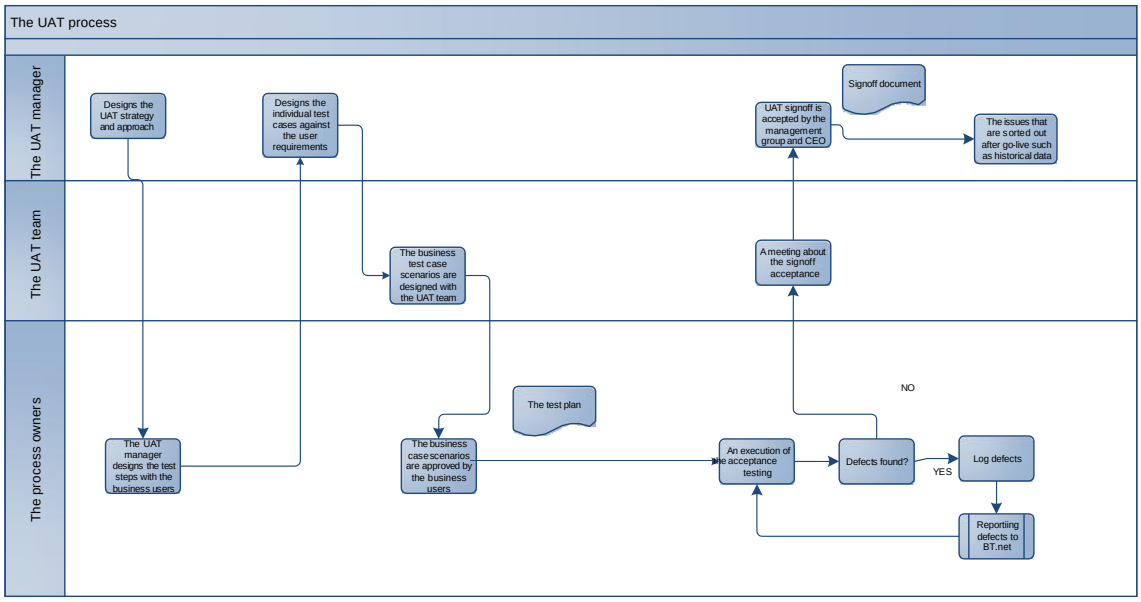



Figure 23. The UAT process in SATEL Oy.

After, acceptance testing is carried out successfully. The UAT will team meet up and agree that all issues have been sorted out. After the UAT test team meeting the steering group of the ERP project gather. The CEO of the company signs off the project. The UAT process in SATEL can be seen in figure 23 above. The UAT process is a combination of theory knowledge regarding the test team and test plan. In addition, the UAT process has some elements from our own operation such as what type of defect reporting system we have in use.

### Designing the sales order process testing

Designing the UAT for the company has not been an easy task. On the contrary it has been very challenging without any previous experience. The most important documents have been the requirement specification and the knowledge of the company's business processes.

The categorizations of the requirements are needed. The requirement specification document can easily be over 100 pages. Different requirements are not necessarily in a neat order for designing the test cases.

It is important that the categorization is done according to the business process, for example, sales order process. It can be done by grouping requirements of the same domain area into their own groups, for example, test portfolio called sales order handling. This grouping must be done for every requirement. Then next step is to assign test steps for the requirements in that sales order handling –group. Test steps are given within that group. This was done with the knowledge gained from the theory presented in chapter 3.4.

This phase can be very difficult to do on your own. It is better to do this phase of the designing with the business process stakeholders, because they have the best expertise and knowledge of their daily work. This was done by personal meetings with each of the business stakeholders. The meetings took 1-2 hours and together with the business process owners the test steps were given to each of the requirements in the domain area.

After this procedure you have all the requirements grouped and the test steps assigned to them, the documents are called acceptance criteria catalogue and test portfolio. The next phase is to design the individual test cases matching each of the requirements in the specification. A comprehensive list of the sales order process with the test steps can be seen in appendix 2. There were no direct instructions how to proceed with this. We applied the information from theory and mapped them into the processes with the help of the business users.

The writing of the test cases is very time consuming. It can easily take a few months to carry out this boring but very necessary task. There are a lot of issues to deal with when writing the test cases.

You must have the necessary requirements grouped and the test steps given to them. Introductions to each of the test cases are needed. It tells the tester the background information of the requirement. This is also known as entry criteria. Then you are required to describe the exit criteria known as expected results. It will tell the tester what is the required outcome for the test case. In addition, the tester should have the possibility to write down achieved results and comments. This was done according to the theory found in chapter 3.3.

Finally, the tester marks whether the test passed or failed at the end of the each test case. Each of the test cases is signed by the tester and it will be accepted by the UAT manager. In case of a failed test case it will be reported into the defect reporting system. Those testing tools must be part of the test plan for this particular reason.

In the previous steps, designing the test cases was accomplished. However it is not enough. The testing can’t be carried out with just the individual test cases. The test cases must be linked with the sales order process.

**SATEL SALES ORDER PROCESS**

These test cases have to run successfully before sales order processing:

Test case INT9500-9503 Web shops orders

Test case FE8000-8002 Handling of the decimals

1. The purchase order from the customer is receiver, this will be transformed into a sales order

   Test case FE3007-3032 Controlling the loan devices

2. Sales assistant checks the content of the order and validity of the customer information, updates an order quantities to TIKE –chart, updates the customer registry if needed

The following test cases must be executed at this point of the sales order process:

1. Test case identifier FE3000-3006 Checking the customer's credit balance
2. Test case identifier FE3000 Customer information management
3. Test case identifier FE3000-3003 Maintaining customer groups
4. Test case identifier FE3000-3002 Checking the company code
5. Test case identifier FE3000-3001 Creating the customer with the model template
6. Test case identifier FE3005-3023 The addresses assigned to the customers
7. Test case identifier FE3000-3004 Customer's addresses
8. Test case identifier FE3000-3007 Maintaining of the delivery terms
9. Test case identifier FE3000-3009 Maintaining of the payment terms
10. Test case identifier FE3000-3008 Maintaining pricing conditions
11. Test case identifier FE3002-3015 Handling of the product sales prices
12. Test case identifier FE3001-3011 Additional texts assigned to the customer information records

3. Sales assistant checks if the content of the customer's purchase order is OK, if not the sales assistant asks for the missing information from the customer.

4. The sales assistant inputs the order information into ERP system and updates the delivery time list

The following test cases must be executed at this point of the sales order process:

1. Test case identifier FE3001-3012 The additional texts assigned to sales orders
2. Test case identifier FE3004-3020 Maintaining parent- and child rows in the sales orders
3. Test case identifier FE3004-3021 The sales order line number
4. Test case identifier FE3003-3019 Calculations of the BOMs
5. Test case identifier FE3005-3024 Calculation of the discounts

5. The production confirms the delivery time to the order according to production planning process.

   1. Test case identifier FE3004-3022 Identifying the new sales order for the verification by the production planning process.

6. The sales assistant sends an order confirmation to the customer.

   1. Test case identifier FE3006-3028 Confirmation of the sales order
   2. Test case identifier INT9000-9010 an confirmation order document that is sent to the customer

7. The sales assistant records the order confirmation to the folders.

8. The production prints out the document called a production order

9. If the delivery time changes the customer is informed by the sales assistant and gets a new delivery date.

Figure 24. Test cases have been linked with the sales order process.

The test cases are linked to the sales order process with the help of business process owner. The business process owner has knowledge in which order the

process is performed. Therefore the business process owner has very valuable information regarding the linking of the test cases. This can be seen in figure 24 above.

The best result can be achieved by personal meetings with the business process owner and to plan a test case linking each process individually. The meetings with the business process owner ensure that the requirements are tested in the right order. This method also increases the successful possibilities of the UAT. The business process owners have played an important part in the designing phase. This was stated, in theory, that the business users have the required knowledge but the challenge was how to harness it. We had to arrange many team meetings as well as personal meetings. This was not stated clearly in the theory.

In addition, the test cases have been written and they have been linked with the sales order process. In other words, the test steps have been designed and the testers have a basic guideline by which order they have to test the requirements with the test cases. It is not possible to give an absolute correct order for the test cases. Many of the test cases can be done simultaneously. The business process owner can't provide a accurate order to the test cases but it is important to identify the requirements that prevent running of the process at all.

### Business scenario testing with the sales order process

The testers need the business scenario test cases. This differs from an individual test case which tests only one requirement against one criterion in acceptance criteria catalogue. The business scenario test case includes the sales order process or other processes as well. The individual test cases are tested in the business scenario test case. The objective is to find out if that system is able to run the process with real world examples. This also reveals and identifies unknown areas that have been forgotten from the requirement specification or in the designing phase of the UAT: This was based on the concept found in (Rice 2009.) but we had to change it and adopt specific data. We had to put it

into action in our special environment. The philosophy was good but we needed to apply different information to the business test case situations.

A business scenario test case is illustrated in table 6. Sales order process: Empower IM orders 3 pcs radio modems, 3 pcs trading goods and 6 pcs accessories. The entire list of different types of business scenario test cases can be seen in appendix 4.

Table 6. An example of a business scenario test case.

| Test case name | A sales order process |
|---|---|
| **Test case number:** | Basic business test case scenario 1 |
| **Customer:** | Domestic customer: Empower IM |
| **Company code:** | 2402138-3 |
| **Products:** | 3 pcs radio modems, material number YM0246 SATELLINE-2ASc with frequency 409,250MHz/12,5kHz/2400/11 bit/FI/0,5W<br><br>3 pcs trading goods, material number YA0106 400-435 Gainflex antennas trading goods<br><br>3 pcs accessories, material number YC0271 CRS-TSU 2m cables<br><br>3 pcs accessories, material number YI0017 I-link 100 MB |
| **Delivery type:** | By post |

Information of the customer's purchase order in table 6 above is used for a business scenario test case with test steps in table 7.

Table 7 A business scenario test case with test steps

| Test step | Test case identifier | Result | Pass / Fail |
|---|---|---|---|
| **41/00** | FE3000-3006 | | |
| **41/5** | FE3000 | | |
| **41/10** | FE3000-3003 | | |
| **41/15** | FE3000-3002 | | |
| **41/20** | FE3000-3001 | | |
| **41/25** | FE3005-3023 | | |
| **41/30** | FE3000-3004 | | |
| **41/35** | FE3000-3007 | | |
| **41/40** | FE3000-3009 | | |
| **41/45** | FE3000-3008 | | |
| **41/50** | FE3002-3015 | | |

(to be continued)

Table 7 (continues).

| Test step | Test case identifier | Result | Pass / Fail |
|---|---|---|---|
| **41/55** | FE3001-3011 | | |
| **41/60** | FE3001-3012 | | |
| **41/65** | FE3004-3020 | | |
| **41/70** | FE3004-3021 | | |
| **41/75** | FE3003-3019 | | |
| **41/80** | FE3005-3024 | | |
| **41/95** | FE3004-3022 | | |
| **41/100** | FE3006-3028 | | |
| **41/105** | INT9000-9010 | | |

A lot of designing work has to be done in order to accomplish testing the sales order process. In the end, all the work done in the designing phase is rewarded when the testing is performed. The UAT must be based on the business test case scenarios. This ensures that the requirements are tested and the missing requirements are more likely to be revealed. Testing is done with the real world cases. In my opinion the business test case scenarios are the most important test cases in the UAT. Many of the unknown requirements are revealed at this point.

The designing of the UAT is not fool-proof; there will be always something that has been forgotten. The main objective is to ensure that the ERP software is able to handle the company's daily business and all the processes related to it. Secondary objective is that business scenario testing tries to uncover these unknown or forgotten items in the designing of the UAT.

### The ERP environment in SATEL

The ERP environment in SATEL consists of 10 virtual servers that are virtualized on the VMware Vsphere virtualization platform. The ERP environment is virtualized completely. The production environment has 2 AOS servers, 2 Microsoft SQL 2012 servers, two active directory servers and 1 SharePoint and 1 enterprise portal server.

The entire IT infrastructure has a maximum fault tolerance based on the hardware. The infrastructure VMware environment has 3 physical hosts and it has vmotion capability. If one host goes offline the servers are automatically transferred into another host without interruption. The ERP user might experience slowing of the system. Additionally, the power supply units, the switches, fiber channel controllers and SAN(s) have been duplicated. The infrastructure is backed up on an EMC Data Domain Backup Restorer device which is based on hard-drives. (Onsite and offsite) The UPS devices have been duplicated as well.

The environment is clustered for maximum reliability by the software. This is done because the ERP system is business critical software for our company, it has to run on 24/7/365 basis. This is the reason for two AOS servers and two SQL servers. Active directory servers provide the possibility to login in with Windows username and password with a single sign-on.

The ERP environment has two standalone servers, one for test environment and another for development purposes. The standalone servers include an AOS server, SharePoint server and enterprise portal in a one virtualized server.

The objective is to conduct testing in an isolated environment, also to develop of the system to an isolated standalone system. The production environment is built at the same time but the testing or the development is not done in the production environment at all. The topology of SATEL's ERP environment can be seen in figure 25 below.

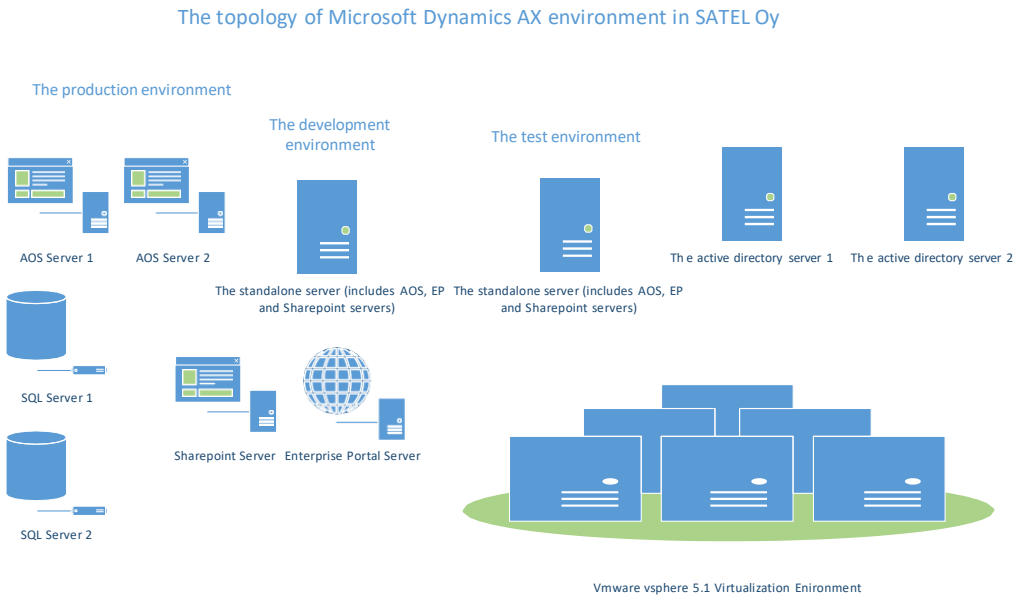



Figure 25. The topology of Microsoft Dynamics AX 2012 environment.

### The defect reporting system

The defect reporting system called BugTracker.net (BT.net is given by SATEL) is used in SATEL for reporting encountered errors or defects during the testing. BT.net is used for controlling the reported defects. This software was chosen because it has contains the needed functionalities for the defect reporting in user acceptance testing. In addition, this software was already used by the R&D department and it is fast and easy to implement into use. It is fully configurable.

The testers report all defects, errors, bugs or questions to the BT.net. The user interface is designed to be very simple and the selection options have decreased significantly. You can see minimal selection options in figure 26 below.

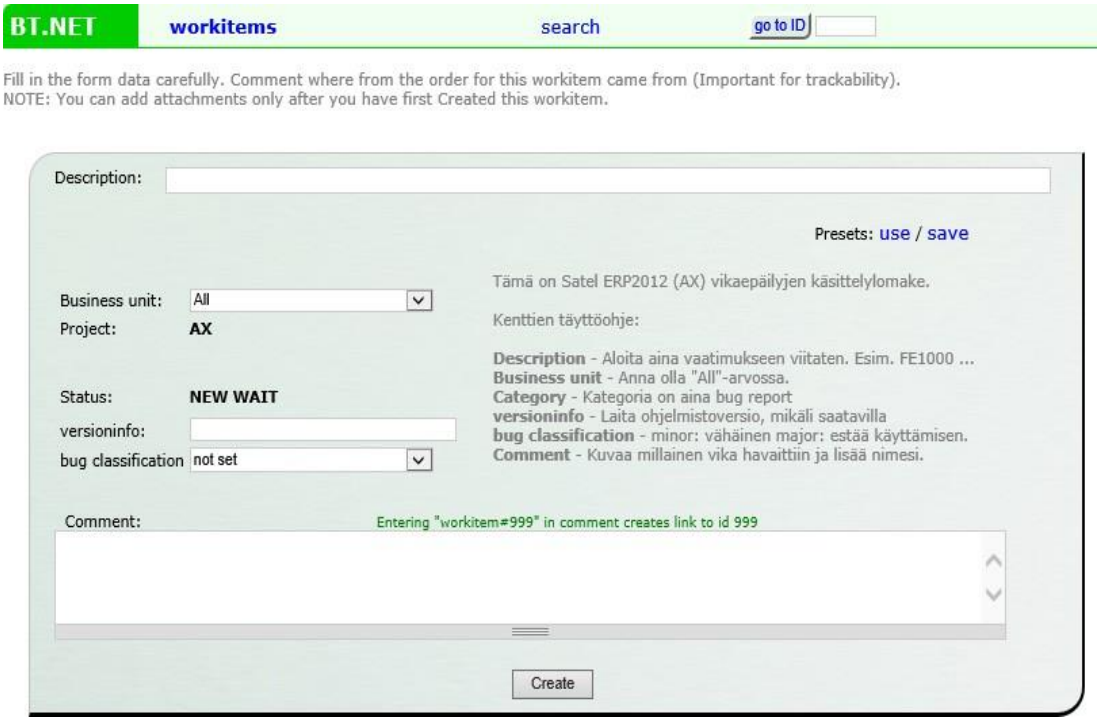



Figure 26. A defect reporting form.

The testers log in to the BT.net by web browser. They can report defects by creating new work items. The site has short instructions on how the reporting is done. In the description field, the tester fills the test case requirement number and the name of tester. This is done because the same user credentials are used by many testers. If a requirement number is not available, then the tester will use the unique number given by the BT.net system. This could happen if there is not a requirement number available for this situation.

The tester only needs to fill the “version info” and the bug classification –fields. Version info field includes the information of which software version the testing was done and the bug classification field tells if it is minor or major. Major bug classification means that it is unable to use the system with that defect. The minor is not a critical defect. Finally, the tester has filled out the defect and reported it by clicking the create button.

The screen captures can be added as attachments into the work item by using Screen Capture software. The tester can easily add any screen captures by clicking the Capture –button.

The tester writes the defect description and instantly the tester can send it. The system will automatically add it as a new work item and adds the screen capture as an attachment to it. At the same time the BT.net system will send an alert email to the UAT manager that a new work item has been added to the system. It waits for the further actions by the UAT manager. Screenshot of the screen capture –software can be seen in figure 27.

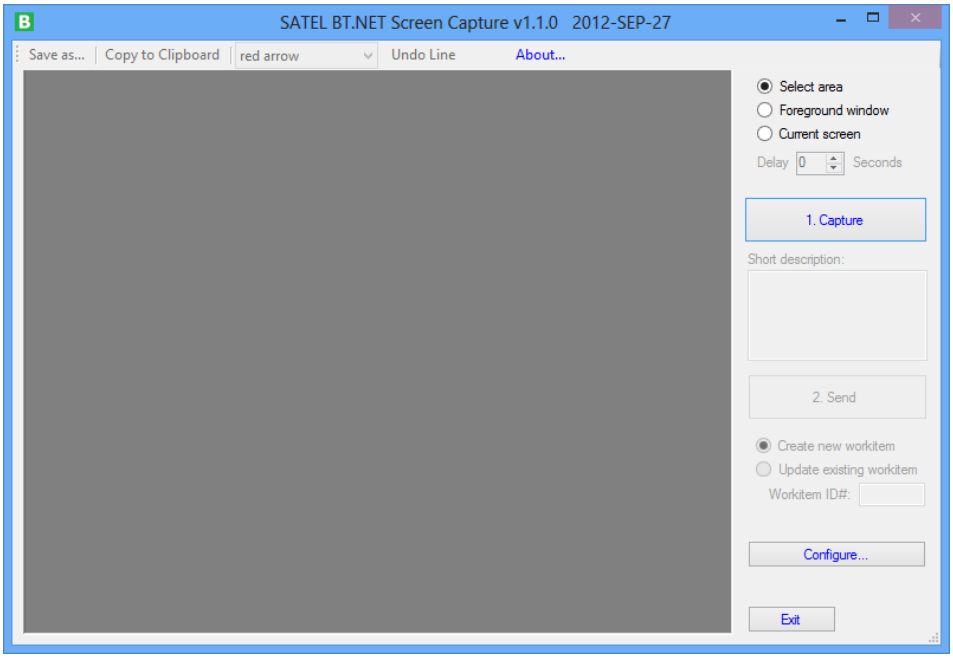



Figure 27. A screenshot of a screen capture program used in SATEL Oy.

The UAT manager can see all work items reported by the testers. In addition, the UAT manager has multiple options such as priority, assigned to, and status field. The UAT manager decides the priority of the defect, and to whom the defect is assigned to and the status of the defect: It can be seen that UAT manager has more options than the testers. The priority can be set and the UAT manager can assign this work item to the supplier or someone else. An example can be seen in figure 28 below.

The defects can be sorted in the defect reporting system by following factors

- User name
- Project
- Organization
- Category
- Priority
- Assigned to
- Status.

This can be seen in figures 28 and 29 below.

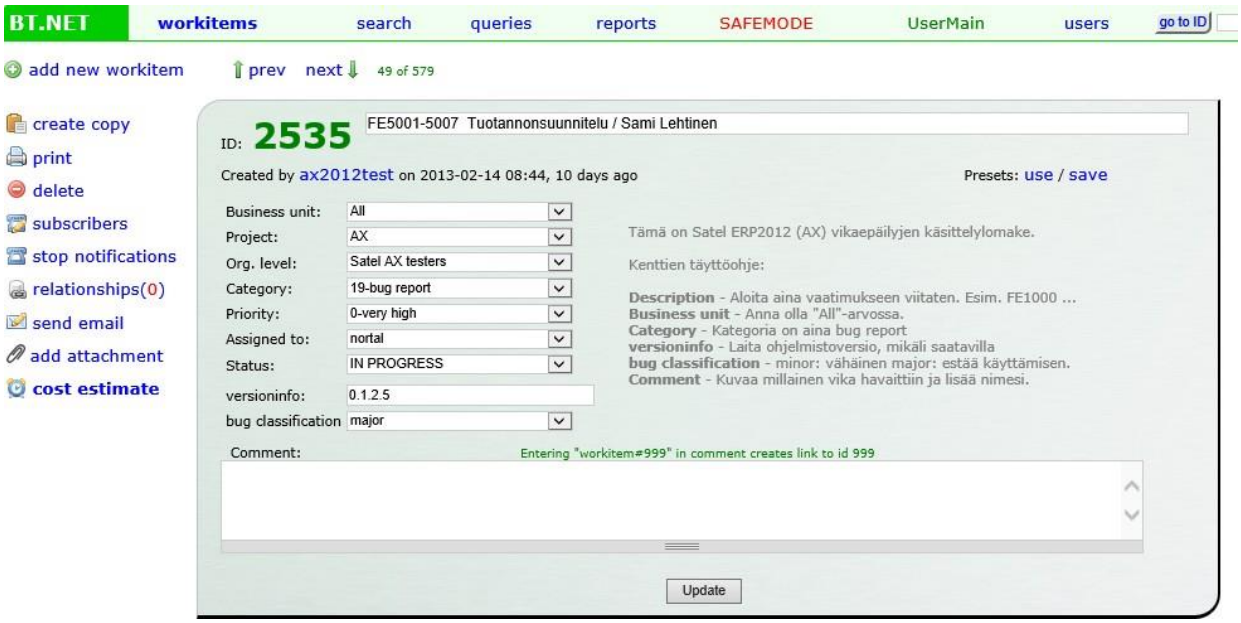



Figure 28. The overview of the defect.


Figure 29. An overview of the defects and their statuses

In table 8 below, are the most common statuses used in defect reporting. The statuses are defined by SATEL. These statuses are used in defect reporting system when the defect is reported.

Table 8. The defect statuses. (SATEL Oy 2010).

| Status | Description |
|---|---|
| **NEW WAIT** | the defect is reported into system but it is not assigned yet. |
| **STUDY** | means that the defect is studied by the supplier/support |
| **WAIT START** | the defect is assigned and waiting for the processing |

(to be continued)

Table 8 (continues).

| Status | Description |
|---|---|
| **IN PROGRESS** | the supplier is working on the defect |
| **PAUSED** | the defect is in paused mode |
| **CHECKED IN** | the defect is reported fixed by the supplier but it waits for the transferring into deliverable |
| **RE-OPENED** | the defect is corrected by the supplier but the SATEL has tested it and it does not work |
| **CLOSED** | the defect is fixed and tested, result OK |

It was decided that only the defects are reported to the BT.net in SATEL. The testing tools such as screen capture software and defect reporting system play an important part of the UAT in SATEL. It is very advisable that similar software is used in the UAT and the proper training is given to the testers.

In addition, test tools should be easy and fast to use with minimum selection options available. The testers' have so many items in mind when they are testing business processes. It is mandatory that the tools are easy enough to use as it increases the ease of the testers. They do not need to worry about using difficult tools with multiple option choices.

The empirical part of the thesis ends here. The empirical work continues in my company by testing according to the user acceptance test plan. The project continues and the plan and the defect reporting system are used on a daily basis. The objective is to take the ERP system into deployment phase with help of UAT plan.

# 7 RESULTS

Learned lessons during the project

The ERP project started with the requirements specification workshops. These workshops were held on weekly basis, nearly 40 times. The ERP vendor interviewed our business stakeholders without becoming acquainted with our current ERP system. This led to serious problems with the requirement specification that was learned afterwards. This led to the situation that our business users forgot to provide critical information about the current ERP or assumed of course the new ERP has similar functions. In addition, the ERP vendor promised that all requirements were possible with the new ERP system. It was discovered afterwards that this was not the case and this would have been easily demonstrated by comparing the current ERP with Microsoft Dynamics AX 2012. This should have been done in a way that the two ERPs are run simultaneously and watching the real differences between the systems. The ERP vendor should have used this method continuously in the workshops. They were concentrating more on interview type workshops. A comparison between the current system and the new system was not directly stated in the theory. This is a very important focus. The theory explained a lot of different types of requirement analysis but they did not cover how much time these phases can take and how they should actually be done.

In addition, the fit-gap analysis done before the ERP implementation project was good but the ERP vendor seemed to have forgotten the important results of this pre-project. If the ERP vendor had done their job accordingly many of our workshops during the project would not be needed. This was a bit of a surprise to us. On behalf of the ERP vendor I have to say that they are still young in the ERP business and they are continuously learning.

The UAT preparations started immediately after the ERP contract was signed but the problems were encountered in that area as well. The requirement specification workshops were unprepared by the ERP vendor and it consumed a lot

of company's resources. This led to a situation that a requirement specification documentation which is the basis of the acceptance criteria for UAT was postponed and the designing UAT was impossible to start. We had to wait for the requirement specification documentation. Because of this designing the UAT started with a draft version. I, as the UAT manager was not happy with this. The UAT theory discusses that you need to start to plan as soon as possible after the ERP project is launched. Using a draft version of the requirement specification made the UAT planning difficult from the beginning.

In addition, when the final requirement specification document was given to us, there were many requirements alongside the following comment "it will be planned during the designing phase." These types of comments in the requirement specification are unnecessary because issues are left open. Writing the test cases according to the requirements and without even seeing the Dynamics AX ERP running when the designing phase of the UAT was run led to challenges. The test cases were written but the problem was that they did not include the instructions for using the system.

It was learned during the project that communication with the UAT team and between ERP vendors are essential part of a successful UAT project. The designing phase of the UAT is the most important and it has to be done together as a team. The UAT manager can't do the plan without communicating with the team and business stakeholders. This was anticipated and based on the information found in Abbaszadeh et al. (2010, 83).

The time schedule of the ERP deployment has been postponed twice. The ERP vendor was excessively positive with the deployment process so they have underestimated the workload of the data conversion and complexity of our business.

There were many times that testers were skeptical towards the entire ERP project. This question was raised many times: "Can we cancel the project?" The reasons for this reaction were unrealistic time schedule of the ERP project. ERP vendor also tried to speed the project two separate times in the steering group.

This affected the UAT team. It was perceived negatively and it raised concerns about the overall project success.

These reactions are quite common when people confront uncertainty and they are against changes. This was seen from the reactions of the users when the testing time was nearing. Users were worrying about the capability of the software and ERP vendor.

One objective with the UAT team is to raise their confidence in the project. In the UAT and in the product these issues were taken into consideration by the UAT manager and Chief Operations Officer. Test execution, resources and time schedule were the worrying issues amongst the UAT team members. Testing time schedule was dependent on the ERP project time schedule. The ERP vendor suggested that the testing should be started in the development environment. That kind of suggestion was denied by the UAT manager because it's against the theory of the UAT. Testing must be done in the dedicated testing environment which is a stable and reliable platform for testing purposes only. This was stated by ISO27001 (2005, 18).

The issues that were worrying users were treated with the solutions before next weekly meeting. The most experienced ERP users that are the UAT team members were taken into the ERP steering group and the training issues were dealt with immediately with the ERP vendor. Training is an important factor in ERP and UAT projects. It should not be underestimated. Using the most experienced ERP users was anticipated. This was stated by Abbaszadeh et al. (2010, 83).

Testing started with the smoke testing. It was soon realized that the environment is not even ready for the testing because smoke tests failed; these sessions were changed into configuration events.

The testing plan has been in use by the finance department and they have the best opinion of the plan. The rest of the business processes are still in the configuration phase and even some configuration days are scheduled on April.

We had representatives of the ERP vendor with us on our testing days. This has been a good approach because the users get used to with the system as well. In theory this approach was not clearly stated. The concept was discussed, having good communication between the customer and the representatives of the ERP vendor. This method turned to be a very good approach.

All the defects, problems have to be reported to BT.net defect tracking system. The training of the test tools was given to the UAT team members at our weekly team meetings by the UAT manager. Testing is proceeding at the moment and the users have used the defect tracking system now by themselves. The financial department has been able to do testing because their work issues are based on law. The business process scenario test cases are another story.

### The opinions about the UAT plan

The UAT plan is the master document for testing. I asked the personnel from the ERP vendor and from our own personnel to give comments about its effectiveness and what should be done differently. The complete answers can be seen in the appendices but I summarize some key points here.

- The plan is useful and testing without it would be difficult
- It is systematic and logical
- Help the developers to understand processes.

The rest of the comments and thoughts can be seen from the appendix 5.

### Recommendations

The companies, who are starting to plan to implement a UAT test, should take into consideration these recommendations. Whoever the ERP vendor is, it is very important that they will be acquainted with your current ERP system. It is a

really important that the ERP vendor sees your business processes in action with the current ERP.

When you are holding requirement specification workshops you should keep in mind that it is unwise to assume that the new ERP system behaves in a similar way as your current ERP. Most of the software can't do what they are assumed to do and the vendor's understanding might differ dramatically from your requirements. Make sure that the UAT is part of your approval process and it is mentioned in the contract as well. It is unwise to try to outsource the UAT. The UAT is your responsibility to make sure that you will receive a system that meets your requirements. This was stated by Windle (2010, 24).

Any changes made by the steering group such as implementing more tailored features or even version change to the newer version can cause unexpected delays that directly affect testing.

Planning of the test cases is very time consuming and even boring work to do. If this phase is done well it will ease the pain of testing. You should start designing the UAT immediately after the ERP project has been launched. It is wise to gather the UAT team from the most experienced ERP users and the users who know the business processes of your company. The meetings should be conducted on a weekly basis. This is the communication channel inside of a company with UAT related issues, it is also advisable to invite ERP vendor representatives to these meetings for better communication at least when the testing of the system starts. In addition, make sure that you have dedicated testing environment and defect reporting system in place. (Windle 2010, 155.)

Time for testing should be reserved beforehand and it should be reserved in mind with the pressures from the executives. It can dramatically effect this scheduling. The fastest way for a UAT to fail is to speed up the test timetable with unrealistic schedules.

You should avoid a situation that the business executives change the UAT time schedule in the steering group meetings. The easiest way to speed up the ERP implementation project is cutting test time. This is ill advised. After testing has

started you should prioritize business process requirements in a way that you are able to start testing. Testing is a very time consuming process. In the case of business process testing, it is a good method that the tester is testing and reporting into defect system is done by another person who executes tests with the tester. In addition, the ERP vendor representatives should be with the testers at least until the first business processes have been tested once or twice.

The ERP vendor consultant should be with the tester when tests are executed. It is a very effective way to do the testing if a problem with the system occurs. Even though all reporting of the problems should be recorded into the defect reporting system, some may not appear on paper. The consultant can give training for the system which is under testing at same time. The theory did not discuss having the representatives available with us when testing is ongoing. This turned out to be a very good approach because testers received training and personal help during testing. On the other hand, it is not recommended to have representatives available all the time. Some testing runs should be done by testers only.

All the changes in the ERP project should be dealt with in steering group meetings. The change control mechanism has to be in place before the ERP project starts. It is good to acknowledge that the ERP vendor might try to speed up your ERP project constantly. This is even done at the expense of testing. The training of your UAT personnel can’t be overlooked. Training is a good way to raise confidence in the UAT.

With these recommendations you should be able to avoid 7 major risks in the UAT such as

- UAT environment setup and deployment process, make sure that you have dedicated environment for development, for testing and for production.
- Make sure that you define your testing objectives in your UAT plan.
- You need to have systematic method to handle the defects.

- Unskilled testers or tester without business knowledge endanger your UAT. Therefore it is advisable to select test team members who have experience in using ERP system.
- Improper communication channel. Make sure that you create a separate UAT team for communication purposes.
- Asking somebody else to perform your UAT. Do not outsource your testing. When it comes to your own business, you are the professional.
- The blame game. You should try to build a positive relationship between business users and the ERP vendor. (Software testing help 2012.)

10 simple steps for taking the UAT into your ERP implementation project are:

1. Ensure that UAT is mentioned in the ERP project contract and the success of the UAT is a requirement for accepting the entire project.
2. Select a UAT manager and the UAT team from the most experienced ERP users. It is also advisable that there are people with different skills.
3. Ensure that the communication channels are clear and precise.
4. All issues that concern timing, costs, tailored features or version upgrades is dealt in the steering group, ignoring this fact will endanger your UAT.
5. Select the testing approach and strategy.
6. Identify the risks related to your UAT.
7. Design the UAT plan and the test cases with the UAT team (remember the most important tests relate to the business processes of the company, the ability to use the new system when the test cases are created is most preferred.
8. Make sure that you have a test environment and defect reporting system up and running.
9. Preserve enough time and resources for your testing.
10. Expect the unexpected. You can’t prepare yourself for every situation that happens in testing.

# 8 DISCUSSION

Summary

The objective of this thesis was to design a testing plan for the ERP implementation for SATEL. How to design a UAT plan? This was the main research question. The constructive research method was used for this thesis. The UAT theory was researched from literature, social media platforms and by contacting UAT professionals. The constructive research method applies to this type of research domain area which is trying to create a new construction for the problem. The plan was made and it has been tested by the finance department.

The UAT plan was designed based on the theory information found in literature. The most challenging tasks were to figure out how to use acceptance criteria catalogue, how to write the test cases and business test case scenarios. In addition, test approach had to be figured out between manual or automatic testing.

The theory explained these issues but they were sometimes too broad and vague. The theory did not give direct answers to my problems. I had to apply and combine the information from various sources. Personal conversations between myself and Mr. James Windle helped a lot to understand the UAT theory.

The ERP project contract was signed in April 2012 after the fig-gap analysis project which took about 6 months to complete before signing the contract. After the contract was signed the work with designing UAT the plan for SATEL started. The UAT plan was written in 3 months in the autumn of 2012. It was designed with the business stakeholders and it was approved by the business stakeholders in the UAT weekly team meetings.

The UAT team was formed in the autumn of 2012. The objective of this team was to share information on problems and challenges of the project and to maintain and control the subproject called UAT project inside our company. The UAT team was formed based on the theory found in (Dustin 2002, 66-70).

Testing started on a weekly basis in February 2013. The ERP deployment is scheduled in the middle of May 2013 but it remains to be seen if we can achieve the goal.

There are challenges despite the testing that has started. There are some business processes that have unfinished configurations and they can't be tested yet. It remains to be seen when these processes are ready to be tested with the help of the UAT plan.

### Limitations and validity

There are limitations in this thesis such as all the test results beyond the scope of this thesis and the UAT plan is specifically designed for SATEL. The UAT plan in this thesis can't be applied to another organization because it is designed for a specific company.

In addition, creating an overall UAT plan is impossible to do. There are always issues that have been forgotten or overlooked. All the business processes have not been documented or the current ERP system did these tasks automatically in the background. In addition, designing the UAT without the new ERP installation was a challenge. This caused plenty of problems in the test case creation phase. This thesis was done with the traditional testing approach. The applicability of the agile methods was beyond the scope of thesis.

The validity of this research is effective because the objective was to answer to the research question "How to create a UAT plan for SATEL?" The information is based on the theory. The constructive research method is based on the theoretical connection. It was possible to construct a UAT plan based on theory from literature, articles and information gathered by other means such as social media platforms. However measuring the effectiveness of the UAT plan in operational level remains uncertain because the UAT plan has actually only been used in the finance process by the financial manager.

The UAT plan is only applicable for SATEL and the company can benefit from the UAT plan results found during the project. Useful information can be found for any organization that plans to do UAT in ERP implementation project. SATEL can learn from this UAT project which parts of the project went well and parts that need improvement. The main result was to design the UAT plan and how it can be utilized in an ERP project. The successful ERP implementation with the UAT plan is an objective as well. In addition, the secondary results such as, communicating with the ERP vendor or insufficient requirement specifications are noted. These are results that the company should utilize in the future.

### Further research

It would be possible to do further research based on the findings in this research for example how to improve customer-vendor communication in a large scale ERP –projects. In addition, further research could be done with the understanding that the UAT plan will affect the success rate of the ERP project.

.

| | |
|---|---|
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

| | |
|---|---|
| | |
| | |
| | |
| | |
| | |
| | |