TALLINN UNIVERSITY OF TECHNOLOGY
Faculty of Information Technology
Department of Computer Engineering
Chair of System Software

IAG70LT

Mihhail Marenkov 132462

Efficiency from using standards for IT projects.

Master`s thesis

Supervisor: Vladimir Viies
Associate professor
Consultant: Taavi Valdlo
Senior Standardization Expert

Tallinn 2014

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõpütöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

_____ _____

(kuupäev) (allkiri)

Abstract

The goal of this thesis is to describe an opportunity of standards implementation in information systems, by giving information about benefits of standards. The standards implementation is based on abstract project and on real information system. Research will show the main standardization opportunities, benefits for IT projects and how standards can assure quality in projects. Moreover, we evaluate standards efficiency by using CMMI quality model, and we will explain our choice. Work covers such important topics as standardization, quality models, quality assessment and assurance. Thesis gives answers on questions why standardization and continuous quality evaluation are vital.

The main conclusions we made relate to the importance of using standardization. For example: global recognition, opportunity to compete, create quality and reliable software. Quality models are created to test, expand and assess the software without stopping the developing process. Quality assessment does not take a lot of time if standard was implemented in right way. In addition such approach can be used to provide greater security and prevent possible failures and errors on earlier project stages.

The thesis consists of four parts. The first part describes goals of standards, which standards are used in Estonia, what is International Standard. The second part provides different standards and methodologies in accordance with software development component. The third part includes implementation of quality assessment model in abstract and real project. Analysis of issues and possible solutions is presented as well. The last part represents the main positive and negative sides of using quality models and standardization. While writing the thesis, we wrote an article on topic “Implementing Quality Assessment Model in IT Project”, which participated in the conference “The 11th International Baltic Conference on DB and IS, DB&IS2014”. Article can be found in Appendix 1.

The thesis is in English and contains 87 pages of text, 4 chapters, 14 figures, 12 tables.

Annotatsioon

Lõputöö eesmärgiks on kirjeldada viis standardite rakendamiseks infosüsteemides, andes informatsiooni standardite eelistest. Standardite rakendamine põhineb abstraktsel projektil ning reaalsel infosüsteemil. Uuring näitab peamisi standardiseerimise võimalusi, eeliseid IT-projektidele ja kuidas standardid võivad tagada projekti kvaliteedi. Lisaks, me hindame standardite efektiivsust, kasutades CMMI kvaliteedi mudelit. Töö hõlmab selliseid olulisi teemasid nagu standardiseerimine, kvaliteedi mudelid, kvaliteedi hindamine ja tagamine. Kõik need teemad peavad olema hoolikalt uuritud igas IT projektis, mitte ainult väikestes infosüsteemides. Lõputöö annab vastuseid küsimustele, miks standardimine ja pidev kvaliteedi hindamine on üliolulised.

Peamised järeldused, mis me tegime, on seotud standardimise kasutamise vajadusega. Näiteks: ülemaailmne tunnustamine, võimalus konkureerida, luua kvaliteetset ja usaldusväärset tarkvara. Kvaliteedi mudelid on loodud selleks, et testida, laiendada ja hinnata tarkvara peatamata arendamise protsessi. Kvaliteedi hindamine ei võta palju aega, kui standard on rakendatud õigesti.

Lõputöö koosneb neljast osast. Esimeses osas kirjeldatakse standardite eesmärke, mis standardeid kasutatakse Eestis, mis on rahvusvahelised standardid. Teine osa pakub erinevaid standardeid ja meetodeid vastavalt tarkvaraarenduse komponendile. Kolmas osa hõlmab kvaliteedi hindamise mudeli rakendamist abstraktses ja reaalses projektis. Probleemide ning võimalike lahenduste analüüs on samuti esitatud. Viimases osas on toodud peamised positiivsed ning negatiivsed küljed kvaliteedi mudelite ning standardiseerimise kasutamisest. Samuti kirjutasime artikli „Kvaliteedi hindamise mudelite paigaldamine IT Projektides“ mis osales “The 11th International Baltic Conference on DB and IS, DB&IS2014” konverentsil. Artikkel on toodud töö lõpus, osas Appendix 1.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 87 leheküljel, 4 peatükki, 14 joonist, 12 tabelit.

Abbreviations

IEEE – Institute of Electrical and Electronics Engineers

ISO – International Organization for Standardization

IEC – International Electrotechnical Commision

ITIL – Information Technology Infrastructure Library

CMMI – Capability Maturity Model Integration

XP – Extreme programming

ISACA – Information Systems Audit and Control Association

CEN – European Committee for Standardization

EVS – Eesti Standardikeskus

GOST – state standard (Russian: государственный стандарт)

IBM – International Business Machines

IT – Information Technology

EU – European Union

EVS/TK 4 – Information Technology Standardization Technical Committee (Estonian: Infotehnoloogia standardimise tehniline komitee)

IS – International Standard

ISMS - Information security management systems

FMEA – Failure Mode Effects Analysis

ISKE – Information Systems security system

ICT – Information and Communication Technology

RIA – State Information Agency (Estonian: Riigi Infosüsteemi Amet)

Content

List of figures

List of tables

Introduction

In Estonia and in Europe there are a lot of electronic services. Estonia has developed ID-card, Mobile-ID and other public electronic services. However the quality of many services has not been measured in detail. Standards specify general principles and recommendations for initialization, implementation maintenance and improvement of information security management. Objectives highlighted in standards intended to execute and implement information security in accordance with the identified risk assessment.

All modern technology and a large part of everyday life are based on standards. They are working in the background and users notice it only when problems arise. Standards setting up requirements for big products or application classes. Standard is a normative document, which is aimed to achieve standardization objectives. The standard is not compulsory, but it is possible to make it mandatory by changing laws and regulations. It is recommended to make standards available to everyone.

Nowadays, there is a lot of high technology, and it includes software systems. Requirements and expectations are changing too. Users are not satisfied with the systems that do the wrong thing, or it is slow and unsafe [1]. Every day IT companies from all over the world develop new projects and this process will never end. The quality of specification, design, code refactoring, software quality assurance becoming more and more important in competitive IT organizations. In recent time more emphasis is put to improve quality using already created tools and techniques for software development [2].

1. Standards in Projects

1.1. Background

Standardization is an activity to establish solutions to a repetitive situation. This is a tool for fixing requirements and improving quality, standards represent universal classes of demands. The choice of possible standards and methodologies is quite large. What to choose depends on the project needs [3].

Some standards, methodologies, techniques and tools for project improvement are very specific and detailed. For instance: ITIL (Information Technology Infrastructure Library), CMMI (Capability Maturity Model Integration), XP (Extreme Programming). Other can be comprehensive (not large), like ISACA guidelines and standards for information system auditing, ISO/IEC 9126, 12207. All these frameworks can be divided into information technology oriented and general. Most popular general are: ISO 9000, Six Sigma, they contain a lot of different tools for improving processes. [4].

Standards can be divided into six groups:

- European standards (CEN, CENELEC)
- International standards (ISO, IEC)
- Local (National) standards (EVS in Estonia, GOST in Russian Federation)
- Organizational standards (within corporations e.g., Google, Samsung, IBM)
- Industry standards (W3C)
- Professional standards (IEEE, INCOSE)

The main standardization concerned parties in IT:

Manufacturer. Company or person who provides IT services wants to create reliable, competitive and high-quality software in the shortest time.

Customers. Customer wants to pay money for good quality, quality is associated with standardization.

End-users. Users who will use a software want to use the soft with a simple and intuitive UI, reliable and properly secured.

Authorities. In countries like Chine or Russia some IT standards are mandatory.

Scientists. IT standards developers and students are interested in new standards.

In the last decade importance and popularity of IT standards increased significantly. Old standards quickly replaced by new ones. Some are not updated, because there is no need in them anymore. And of course, due to software and technology development new standards appear very often (International Organization for Standardization created 1103 standards in 2013, which is 58793 pages [5]). In the table below are shown standards, which have been replaced in last time:

Table 1. Replaced IT Standards

Old Standard	New Standard	Current status
ISO/IEC 9294:2005 Information technology – Guidelines for the management of software documentation	ISO/IEC/IEEE 26511:2011 Systems and software engineering - Requirements for managers of user documentation	Revised
-	ISO/IEC 29146 Information technology - Security techniques - A framework for access management	Under development
ISO/IEC 18045:2005 Information technology - Security techniques - Methodology for IT security evaluation	ISO/IEC 18045:2008 Information technology - Security techniques - Methodology for IT	Revised

	security evaluation	
ISO/IEC 9995-8:1994 Information technology - Keyboard layouts for text and office systems - Part 8: Allocation of letters to the keys of a numeric keypad	-	Withdrawn
ISO/IEC 29881:2008 Information technology - Software and systems engineering - FiSMA 1.1 functional size measurement method	ISO/IEC 29881:2010 Information technology - Systems and software engineering - FiSMA 1.1 functional size measurement method	Revised

So, some technologies may eventually become not demanded, in this case, because of no need standard will be withdrawn. If appears a new, stable technology, when big standardization companies start writing a standard for this technology, and after revise if it will be needed. In my opinion, this is an endless loop.

1.2. Goals of Standards

The main goals of standards are:

- Minimizing expenses
- Ensuring the competitiveness and product quality
- Improving the level of security for human life and health, the environment and property
- Ensuring the uniformity of measurements
- Ensuring the rational use of resources
- Finding optimal solution. Identifying the most appropriate and cost-effective option

Summarizing, it can be said that the goal of standardization is ensuring the competitiveness and product quality, minimize expenses, improve the level of security for human life and health, ensure the rational use of resources, and find optimal solution and many others [6]. In addition to that, company what is using standards knows what

to wait from IT, before project starts, it knows what users are waiting from IT and standards also help understand, how IT services from one company can be compared with other services.

1.3. IT standardization in EU and Estonia

Estonia is using a series of IT standards, which include international experience. Most of the standards are translated. Also there are created information technology rules for Estonian language and culture, information technology dictionary (about 40 parts) and other original materials.

Estonian IT standards can be used:

- Designing, managing and organizing software related workflow (ISO/IEC 12207, ISO/IEC 15288)
- Quality management (ISO/IEC 25000, ISO/IEC 90003)
- Information security management (ISO/IEC 27000)
- IT service management (ISO/IEC 20000)
- Requirements on IT in Estonian language and cultural environment (EVS 8)
- Information technology vocabulary (ISO/IEC 2382)
- IT guidelines for the documentation (ISO/IEC 26514, ISO/IEC 9294)

Standardization regulation in Estonia:

- Estonian Centre for Standardization (EVS)
- For development specialty standards, signed cooperation agreements with professional associations and agencies
- Technical Committees (over 40): EVS/TK3 Telecommunications, EVS/TK4 Information Technology

- Over 24003 of EVS standards, most of them are translated, less than three hundred are original (275)

Technical regulations and standards act:

- Entered into force 01.04.1999
- Contains information about technical norms and Estonian standards
- Using of standards in Estonia is not mandatory

EVS TK4 – IT standardization technical committee

- Founded 29.10.1997
- Over 20 members (AS SEB Pank, Cybernetica AS, Helmes AS, Microsoft Estonia, Swebbank)
- Scope: IT, geography information, building automation, health informatics, road transport and traffic telematics, advanced manufacturing technologies, machine-readable cards

The Estonian Information Technology Standardization Technical Committee EVS TK4 cooperates mainly with International Organization for Standardization (ISO) and the International Electrotechnical Committee (IEC) created by JTC1 committee and the European Standards Organization CEN of information technology (CEN/ISSS).

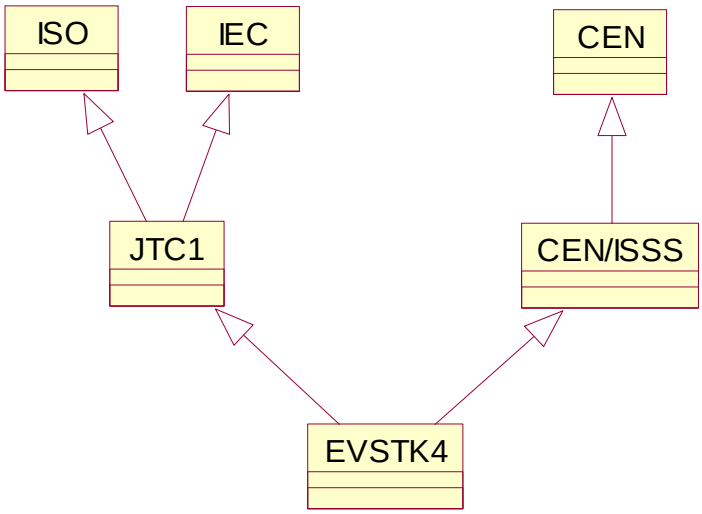

Figure 1. Connection of Estonian and International standards

1.3.1. International standards

Those standards are developed by international standards organizations (ISO) in order to eliminate technical barriers in trade and to harmonize requirements for products and services in line with international standards.

International standards can bring a lot of efficient points into every IT project for many concerned sides.

For enterprises that produce IT services, the widespread acceptance of International Standards (IS) means that suppliers can develop and offer services that meet the technical requirements, which received international recognition in relevant sectors. Thus, IT companies that use International Standards can compete on a larger number of markets around the world. Another advantage is cost savings, because IS are optimizing operations, simplify them and make them more efficient. Also they bring better productivity and competitiveness.

Despite the fact that IS are not mandatory, every government can make a standard as a requirement. For government IS providing the technological and scientific bases in support of legislation in the different fields of IT security. In such way government can

have an expert opinion, without using their services directly. Also this will help to regulate the requirements for import and export.

For consumers IS conformity of services ensures their quality, safety and reliability. So they will be confident that they are safe and have a high quality product.

1.4. Overview of software standards

Software standards classifications:

- General (related to whole life cycle) / organizational / technical (related to some life cycle step)
- International / national
- Formal / non-formal (ISO standards / flow diagrams)
- Organization internal / external

In fact, there are much more standards and their number is constantly growing. Below are the topics on which are created many of standards.

Table 2. Standards and frameworks by topic

Topic	Standard
Terminology / Vocabulary	ISO 8402 EVS-ISO/IEC 2382
The system life cycle	ISO/IEC 12207
Quality Management	ISO 9001 Six Sigma ISO/IEC 15504 (SPICE)
Documentation	EVS-ISO/IEC 26514:2008 EVS-ISO/IEC TR 9294:2003 ISO 15489:2001 ISO 5127
Testing	ISO/IEC 12207

Risk management	ISO 31000
Information Security / Reliability	ISO/IEC 27033 ISO 27001
Testing Documentation	ANSI/IEEE Std 829
User Documentation	IEEE Std 1063 ISO/IEC/IEEE 26511:2011 ISO/IEC/IEEE 26512:2011 ISO/IEC/IEEE 26514:2008 ISO/IEC/IEEE 26515:2011
Auditing	COBIT ISACA ANSI/IEEE Std 1028
Software Requirements Specifications	IEEE Std 830

Analyzing the table above it can be said that there are a lot of different standards and they cover a variety of topics in software development. Depending on the size, complexity, chosen methodology or purpose (international or local) of the project it is possible to select the appropriate IT standard.

1.5. Software documentation

It is important to notice, that there are no uniform requirements for software documentation. There is no single list of documents that is needed to make the development of software. The reason is that software and environment where it will be used are always different. Consequently development process, actions and documentation for each action are different. For example, database documentation cannot be compared with satellite management system documentation. However, there are quite a lot of standards, practices and suggestions on what type of documentation should be created. Documentation for software and other types, it is possible to use following standards (operating in Estonia):

- EVS-ISO/IEC 26514:2008 Systems and software engineering – Requirements for designers and developers of user documentation
- EVS-ISO/IEC TR 9294:2003 Information technology. Guidelines for the management of software documentation.
- ISO 15489:2001 Information and documentation
- ISO 5127 Information and documentation (Vocabulary)

There are also series of standards and recommendations on how should look different software process and activities documentations.

Documentation is very important for every small and large IT project and standardized documentation can bring a lot of benefits to organization:

- Customer can better understand the power of project. Understandable documentation unlocks the potential of project by explaining how to use its features. Many modern IT projects contain some great features that users cannot use because they do not understand how it works (lack of documentation).
- It can bring value to project. Well-structured documentation is an essential part of every project. If product was designed very well, when it is necessary to use standards to documentation as well.
- Opportunity to sell more. Standardized documentation increases customers loyalty and gives a chance that they will buy services again. Also it suits for prospective customers, because it can give a clear idea of how product can be use and can be achieved using a product.
- Gives confidence in quality. Probably the main idea of every standard. Clients very often judge the quality of a project by reading documentation, and it determines the first impression.

1.6. Standardization control, Audit

An information systems audit is an expertise of the management controls within an IT infrastructure.

The main goals of audit can be shown using a use-case diagram:

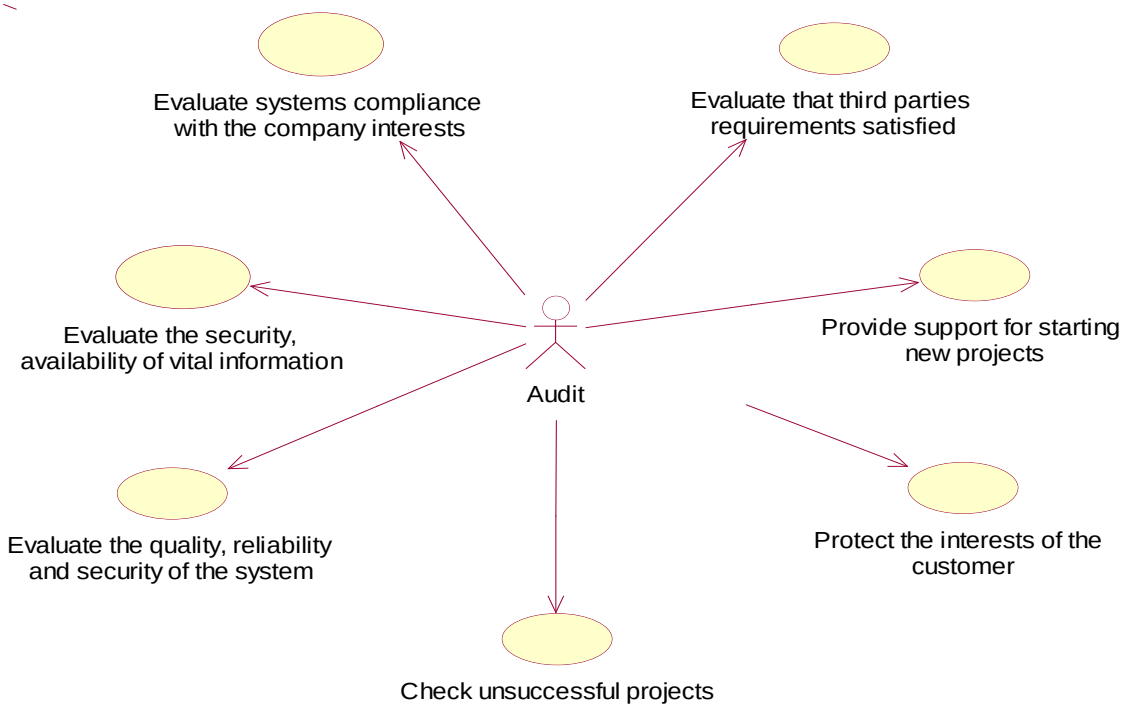

Figure 2. Audit goals

Audit affects all information system related objects, actions, processes and fields, including planning , organization, documentation, project management, development, methodologies, use, maintenance, measurement, reporting, internal quality management.

A variety of demands are presented to the auditor. He should be independent from the project or organization and should be an expert in information systems auditing and in information technology field, monitor good practices and rules of auditing, to be familiar with local (Estonian) legislation and standards and get to know some of the recognized auditing methodology.

The main audit steps are shown in the activity diagram below:

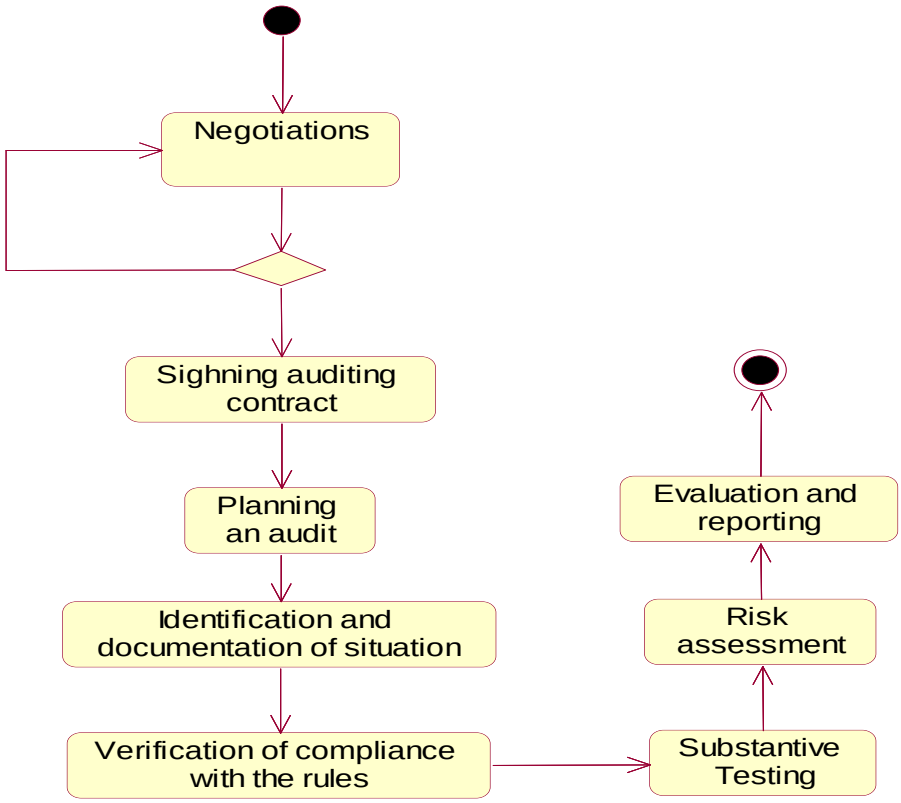

Figure 3. IT auditing steps

While planning, an audit is defining critical areas, holding negotiations about work order and coordination, verification methodology, reports, timelines and workload.

Identification and documentation of situation can be for example development policy, compliance with legislation and standards.

Information system audits from all over the world are united thanks to "Information Systems Audit and Control Association" (ISACA). The association has more than 110,000 members in 180 countries, including Estonia (Estonian Association of Information Systems Auditors). ISACA certifies information system auditors, publishes IT audit literature, develops audit methodologies, organizes courses, carries out researches and development works. COBIT 5 is the largest edition of ISACA globally

accepted framework, which provides opportunities for checking the quality of IT project.

2. Standard/Methodology in accordance with software development component

IT standards can bring a lot of benefits for projects. Standards can be used for every project step or only for one project part. Nowadays, standard is a tool for the implementation of best practices and more and more organizations implement them to avoid risks or to improve security.

There are thousands of standards which are provided by different organizations. For instance ISO provides 19,500 different international standards. Therefore, before choosing a specific IT standard it should be carefully analyzed and compared with other standards.

2.1. Requirements

Requirements can be implemented using different standards. In this part will be analyzed most popular of them.

Software requirements should be correct, unambiguous (simply understandable), complete, consistent, ranked for importance, verifiable (should be opportunity to test a requirement), modifiable, traceable. Most of those rules for requirements can be performed using different standards and specifications.

2.1.1. IEEE Std 830 Practice for Software Requirements Specification

This standard perfectly suits for IT projects. Basing on the standard, the requirements specification can have the next structure and logic.

1. Introduction
   1.1 Purpose
   1.2 Scope
   1.3 Definitions, acronyms
   1.4 References
   1.5 Overview
2. Overall description
   2.1 Product perspective
   2.2 Product functions
   2.3 User characteristics
   2.4 Constraints
   2.5 Assumptions and dependencies
3. Specific requirements

The basic issues that the requirements writer should address are the following [7]:

a) Functionality. What is the software supposed to do?
b) External interfaces. How does the software interact with people, hardware of the system, other hardware or software?
c) Performance. What is the speed, response/recovery time, availability?
d) Attributes. What is the correctness, portability, security, maintainability considerations?
e) Design constraints imposed on an implementation. Are there any resource limits, policies for databases/software/hardware, language?

2.1.2. ISO/IEC 27001 Information security management systems – Requirements

The purpose of this standard is to provide selection of security tools which meet all requirements and help protect information assets and give confidence in the company safety. Requirements presented in this International Standard are common and applicable to all organizations, regardless of type, size and activity. Company should enter, perform, use, control, review, support and improve documented ISMS within the entire business activities of the organization, as well as the risks that it faces.

Standard provides the next requirements for information security management systems:

a) Considering the characteristics of the organization (location, assets, technology) determine the scope and boundaries of the ISMS.
b) Develop an ISMS policy that includes a system of setting tasks, actions relatively information security, business and juridical requirements, obligations for safety, risk management environment.
c) Develop the concept of risk assessment in the organization. Identify a risk assessment methodology. Develop criteria for accepting risks and determine acceptable levels of risk.
d) Identify risks. Identify vulnerabilities in security.
e) Analyze and evaluate risks. Assess the damage to company, which may be caused due to system failures. Consequences of privacy violations.
f) Identify and evaluate tools for reducing risk.
g) Choose tasks and controls to reduce the risks.

This cycle can be repeated more than once. Before this concerned parties should create requirements and expectations of information security. As a result they will have controllable information security. Simplified process scheme:

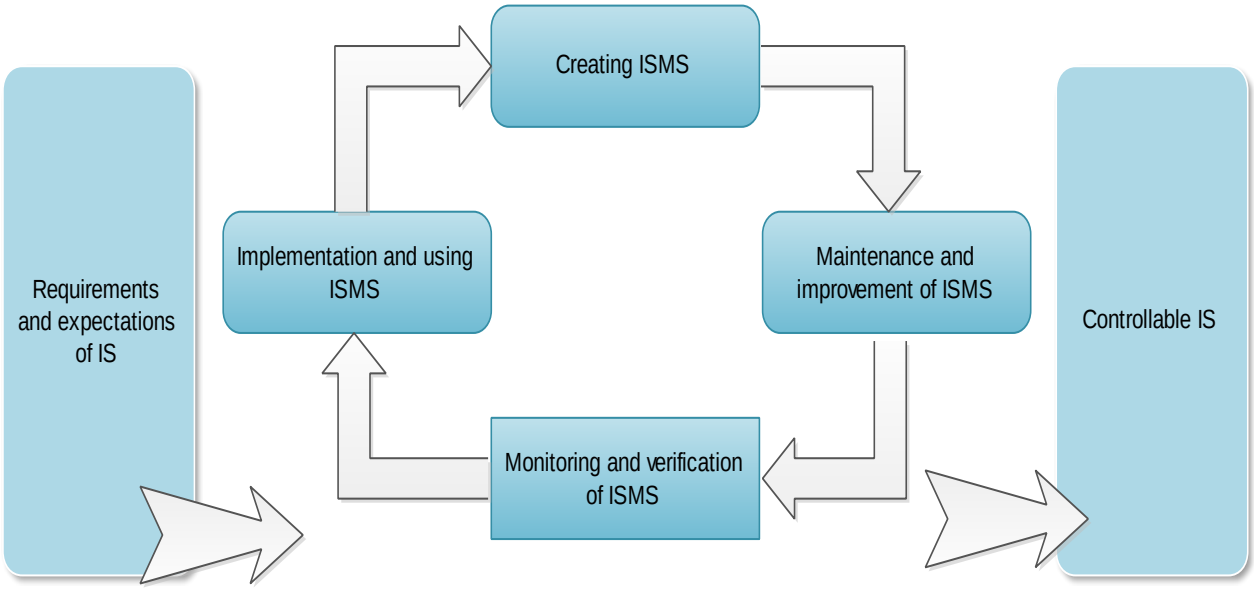

Figure 4. ISMS scheme

2.1.3. ISO/IEC 25010 Systems and software Quality Requirements and Evaluation

Provides list of the quality attributes which can be used to make sure that nothing left missed. Considering all requirements and needs of project, product quality model is shown in the table below:

Table 3. System and software quality requirements

Functional Suitability	**Reliability**	**Performance Efficiency**	**Operability**	**Security**	**Compatibility**	**Maintain-ability**	**Transferability**
Suitability	Recoverability	Resource-Utilization	Understand-ability	Confidentiality	Replace ability	Modularity	Portability

Accuracy	Fault Tolerance	Time-Behavior	Helpfulness	Integrity	Co-existence	Reusability	Adaptability
Interoper-ability	Availability		Attractiveness	Accountability	Interoperability	Analyzability	Install-ability
Security			Learnability			Change-ability	
			Technical accessibility			Testability	

This list of life cycle requirements makes product quality better. It helps to determine project weaknesses and get rid of them as soon as possible.

Summarizing this part, we can make a consequence that most of requirements standards have a very similar structure and they are targeted to perform a one goal. All of them can help to identify necessary attributes, characteristics and successfully play an input role into the design stages of software development.

2.2. Risk analysis

Risk analysis should be done as a part of the risk management in every IT project. The risks of the Information Technology environment are a subject of many methodologies and standards. Considering information from standards it can be said, that information security risk is a potential threat of vulnerability exploitation of property thus causing harm to organization.

Another problem is that sometimes IT risks are overlooked, because leadership mainly aims to prevent market, credit and other business risks.

2.2.1 ISO/IEC 27005 Information security risk management

This standard is designed to determine an approach to the management of risks in organization, depending on field of application or industry sector. The methods

described in this standard, meet the general concepts, models and processes specified in ISO/IEC 27001. These recommendations are intended to help realize information security, based on the risk management approach.

Analyzing this standard, it can be said that risk management contains a lot of different steps and problems. The best way to represent this process is using a diagram.

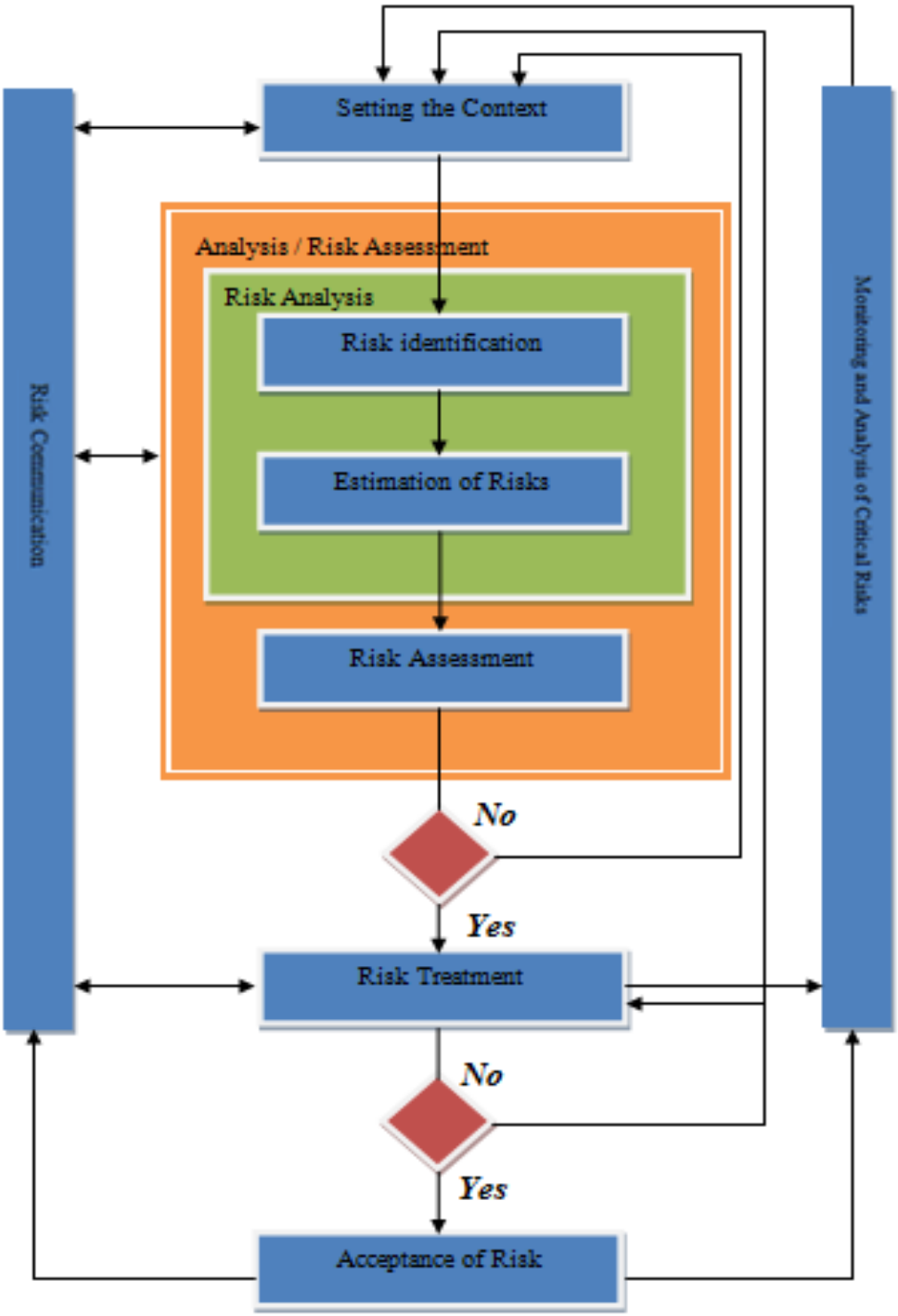

Figure 5. Risk management process

Activities shown in this diagram can be performed more than once. Analyzing this standard it is possible to make a conclusion that an iterative approach to risk management can increase the depth and evaluation details for each iteration.

2.2.2. ISACA RiskIT

Another opportunity to manage risks in IT projects is using RiskIT framework. It helps to understand the most dangerous IT risk types and to make risk-aware decisions.

Considering this framework, every risk (e.g., strategic, environmental, market, credit, operational, compliance) is an IT-related risk. In addition, some risks can reduce value or performance of IT project [8].

In the next table are shown examples that prove that most of the risks have an IT component.

Table 4. IT related risks

Risk	IT Risk	Example
Strategic	IT Benefit enablement risk	Missed opportunity to use more effective IT technology
Environmental	IT Benefit enablement risk	Using IT technologies which can harm an environment
Market	IT project delivery risk	Losing a position on market, because of low quality
Credit	IT project delivery risk	Poor IT security can lead to lower credit ratings
Operational	IT service delivery risk	Inappropriate employee actions bring to not reliable project

Compliance	IT service delivery risk	Costumer gets product, which does not meet the requirements

IT risks always exists, even if they are not recognized by an enterprise. Every business decision needs the manager to balance risks. The Risk IT framework allows users to:

- Integrate IT risk into the "Enterprise risk management"
- Allows to make informed decisions about the extent of the risk
- To be prepared how to respond to the risk
- Increase information quality
- Helps to understand significant IT risk types

2.2.3. Failure Mode Effects Analysis (FMEA)

FMEA is an analysis of the causes and consequences of failure. This method of analysis is used in quality management to identify potential defects and their causes in the product, process or service. It is used to detect problems before they occur and have an impact on the consumer.

FMEA analysis is performed in the following order:

1. Selects an object of analysis. If the object is composite, it is necessary to determine its limits. For example, for analyzing a process part, it is needed to identify the initial and the terminating event.
2. Identification of analysis options. For instance, from largest parts to smallest or vice versa.
3. Define the boundaries within which risk can exist. Boundaries may be – time period, type of consumer, certain actions.
4. Creating a suitable table for registration information. It can vary, depending on considered factors.
5. Defining elements, which may cause failures. If the list of elements becomes too large it is necessary to reduce FMEA boundaries.

6. For each element from previous step creating the list of the most significant failures. This operation can be simplified if company is using a standard list of failures for the elements.
7. For each failure, identified in step 6, determining all possible consequences that may occur. This operation can be simplified if company is using a standard list of consequences.
8. Determining severity rating for the consumer (S) – Severity. Severity rating is usually determined on a scale from 1 to 10, where 1 means low consequences, and 10 disastrous.
9. Identifying all potential causes for each failure mode. All potential causes for each failure mode are brought into the FMEA table.
10. Determining occurrence rating for each cause (O) – Occurrence. Occurrence rating is usually determined on a scale from 1 to 10, where 1 is extremely unlikely event, and 10 means inevitable event. Rating results are brought into the FMEA table.
11. Identify current process control for every cause. For example, tests and procedures which will help to keep failures from reaching the customer. This should prevent the cause from happening, reduce the likelihood that it can occur or detect failure after the cause has already happened but before the customer is affected.
12. Determine detection rating for each control method (D) – Detection. It is rated on a scale from 1 to 10, where1 means that the control can precisely detect a problem, and 10 means that it cannot detect the problem (or control does not exist).
13. Calculating the risk priority number (RPN). It is equal to S * O * D. This number provides guidance for ranking potential failures by relevance.
14. Determine the recommended actions, which can include modifying a project or process to reduce the severity or occurrence of failure. Also there may be additional control measures to increase failure detection.
15. After performing the recommended actions rating values S, O, D and RPN number are recalculated.

To find out which standard can be better for a small information system, we made a comparison table (choosing qualities, which are especially important for every document), which is presented Table 7

2.3. Documentation

Documentation is an essential condition for the successful implementation of any software development project.

2.3.1. ISO/IEC 9294 Guidelines for the management of software documentation

This standard provides guidance on the documentation of software for those managers who are responsible for the products. Guidelines are intended to assist managers in ensuring the effective implementation of documentation in their organizations. It is intended to apply to all types of software – from the simplest to the complex software systems. Covers all types of software documentation relating to all stages of the software lifecycle.

This standard does not provide specific rules for the documentation. Analyzing the standard, program documents can be divided into three categories:

1) Development documentation. Documents describing the software development process, define the requirements to be met by software, determine how it is controlled and how to ensure its quality. Also it includes a detailed technical description of the software (program logic, relations, formats and data storage options). It can be said that this step describes a history of software development.
2) Product documentation. Provides the information, which is needed for maintenance, modernization and transformation. It makes it easier for programmers, who did not develop software take part in modernization.

3) Project management documentation. Based on the project management information. Such as reports on the schedule changes, reports on decisions related to the development, allocation of responsibilities.

2.3.2. ISO 5127 Information and documentation – Vocabulary

The standard provides definitions from more than thousand scientific and technical concepts. The goal of this standard is to unite international professional communication, both oral and written. Difficulties can create language, different cultures, traditions, information and documentation practices, legislation difference.

The terms and definitions are divided logically into seven groups:

- Basic and framework terms (97 terms)
- Documents, data media (173)
- Documentation institutions (66)
- Documentation process (318)
- Use of information and documents (75)
- Preservation of documents (236)
- Legal aspects of information and documentation (125)

2.3.3. EVS 8:2008 Requirements on IT in Estonian language and culture

Not all projects are designed for an international level. Especially, this is much better if the project is documented in accordance with both: international and local standards. Amount of clients in IT projects will grow if organization will understand and respect other cultures and their local standards.

Some examples of EVS 8 standards:

Table 5. EVS 8 Standards

	EVS:8 (Domestic)	**International**
Monetary formatting	1,00	EUR 1,00
Alphabetical sorting	Added Estonian letters after 'w' (w õ ä ö ü x y)	ISO/IEC 10646 (w x y)
International country code	+372	ITU E.164
Country code **Two-letter code** **Three-letter code**	233 EE EST	ISO 3166
Language code **Terminology** **Librarianship**	"et" "est" "est"	ISO 639-1 ISO 639-2 ISO 639-3

2.3.4. ISO/IEC 26514:2008 Systems and software engineering – Requirements for designers and developers of user documentation

The standard recommends that the development of user documentation was a part of a software product development and followed the same processes related to the life cycle of software. Describes how to establish what information users need, how to determine the way in which that information should be presented to the users, and how to design the information and make it available. This standard can be useful for developers of the next types of user documentations:

- Documentation about the product, which is different from software documentation.
- Multimedia systems with animation, video and sound.
- Computer-based training programs and special educational material.
- Service documentation describing the operation of the system software.
- Documentation within the user interface.

There is no doubt in efficiency of this standard, because user documentation is a very important part of every IT project. However, many companies do not use documentation at all. And nevertheless, sometimes there is no doubt of its necessity, because someone will use this software or even more someone will modify it.

In addition in last time appear pretty much new standards (Table 6), which are connected to software documentation and are from the same series as standard described above.

Table 6. User documentation standards

Standard name	Full name	Description
ISO/IEC/IEEE 26511:2011	Systems and software engineering – Requirements for managers of user documentation	Describes the first steps in creating user documentation. Covers management activities in the beginning of project, describes opportunities to set up procedures and specifications, establish infrastructure, and build a team. Contains examples of roles, which are needed in user documentation team.
ISO/IEC/IEEE 26512:2011	Systems and software engineering – Requirements for acquirers and suppliers of user documentation	Created to supply software user documentation as a part of the software life cycle process. It gives directions for setting up requirements for software user documentation.

ISO/IEC/IEEE 26513:2009	Systems and software engineering – Requirements for testers and reviewers of user documentation	Created to provide an accurate, well-structured and complete documentation for software users. Gives recommendations for testing and reviewing the user documentation. Can be used for all documentation types, but requires knowledge of testing.
ISO/IEC/IEEE 26514:2008	Systems and software engineering – Requirements for designers and developers of user documentation	Described above in part 2.3.4
ISO/IEC/IEEE 26515:2011	Systems and software engineering – Developing user documentation in an agile environment	Can be used in companies which are using agile development. Agile methodologies do not provide any guides about documentation; therefore this standard can be very effective.

2.4. Testing and Privacy

The main goal of software systems testing is to help company build quality into the system during the life cycle process and to check that needed quality was achieved. Test process is needed to determine whether the products match all requirements and whether the product satisfies user needs. This process can include inspection, analysis, demonstration and testing of products.

IT projects data should be confidential, available and integral. Information security is the process which is aimed to ensure those components and find a balance between them.

Another question, on which I will answer in this work, is why projects need an IT Security Standards? In this part will be analyzed standards which have appropriate security goals and level, are regularly updated and are free or with low cost.

2.4.1. ANSI/IEE Std 829 Standard for software and system test documentation

This standard gives a structure for test documentation, was the first testing standard. It provides a common framework for testing processes, activities and tasks. Also this standard helps to define the test tasks, required inputs and outputs. In addition to that, standard helps to define contents of test documentation.

Below is the standard structure. Described documents have a different weight. The standard describes the following documents (every document is described in detail and contains a lot of steps and requirements):

1) Test Plan. Identifies test items, features (to be/not to be tested), tasks, who will do each task. Creating item pass/fail criteria. Preparing environmental needs (hardware, operating system, communication requirements). Setting up responsibilities by showing who is responsible for what part. Satisfaction of staffing and training needs. Creating a schedule, risks and contingencies.
2) Test-Design Specification. Writing down what features will be tested, for example the quality criteria and sub-criteria. Includes the organization of the tests into groups. Determining what methods for what tests will be used.
3) Test-Case Specification. Input output specification. Determining what to do in what sequence.
4) Test-Procedure Specification
5) Test-Item Transmittal Specification. Writing down, in what version the failure was found?
6) Test Log
7) Test-Incident Report. Writing down expected, actual results, anomalies, procedure steps, date and time, testing environment and failure impact.
8) Test-Summary Report. Summary from the whole testing process and test results.

2.4.2. ISO/IEC/IEEE 29119 Software Testing

In September 2013 an internationally agreed set of standards for software testing ISO/IEC/IEEE 29119 was developed. There are currently five parts in this standard, a more detailed description of the standard can be seen below [9]:

Table 7. ISO/IEC 29119 standard parts

ISO/IEC 29119-…	Description
1: Concepts & Definitions	The aim is to create an understanding about all other standards in the series. Introduces the vocabulary on which are based other 29119 standards. Provides definitions, description of the software testing concepts, opportunities to apply the processes.
2: Test Processes	The aim is to define a common process model for software testing that can be used in any software development life cycle. Defines test processes, which can be used for management and implementation software testing in any organization.
3: Test Documentation	There are brought documents that are defined in standard. Organizational: test policy, organizational test strategy. Test Management: test plan, status report, and completion report. Dynamic Test Process Documentation: design-, case-, procedure specification, data and environment requirements and readiness report, actual

	report, test result, expectation log, incident report
4: Test Techniques (Under development)	The goal is to create a one standard which will cover test design techniques that can be used in test design and implementation processes. Covered techniques can be found in Appendix 2 Figure 11
5: Keyword Driven Testing (Under development)	This standard helps to describe a test case by using predefined keywords. Such approach will help to easier understand, automate and maintain test cases.

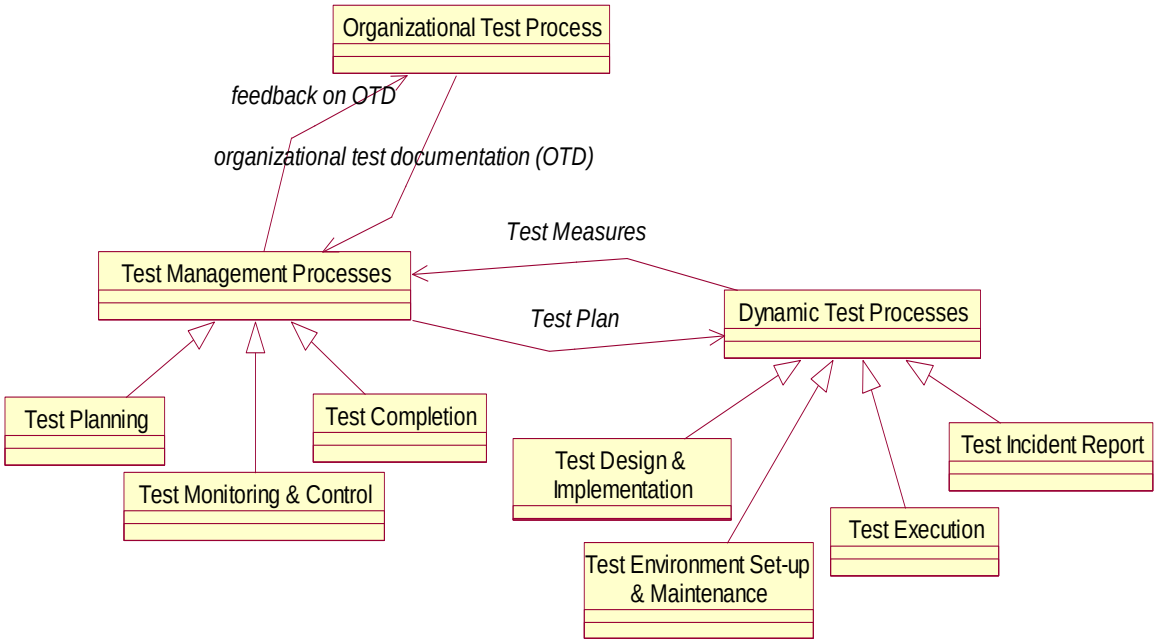

Figure 6. ISO/IEC/IEEE 29119 Test Process Model

This set of standards is very detailed and it does not insist on using every part of it. Considering all analyzed data it can be said that using of all standard steps and techniques can be necessary only for big and serious projects, for small IT projects there is no need in using all of the standard benefits.

2.4.3. ISO/IEC 29101:2013 Security techniques

Important to notice, that this is the first information security standard in the world, which was developed by Estonian editors. Estonian experts made a very important contribution in development of the standards content.

Standard defines a privacy architecture framework that determines concerns of information and communication technologies. Also, standard defines components and aspects for implementation ICT systems and provides architectural views.

Standard contains:

1) Guidance, how to customize terminology, what is related to private data to information systems.
2) Software architecture views, which help architecture manager to plan the system.
3) Description of personal data protection software components.

2.4.4. ISKE implementation

For protecting Estonian State and local government databases, there must be implemented three-level baseline system (compulsory, according to Government Regulation no. 273). This means, that there should be three different sets of security for three different security requirements. ISKE is based on the IT baseline protection handbook (IT Grundschutzhandbuch) which is published by the German information Security Agency (Bundesamt für Sicherheit in der informationstechnik, BSI). BSI system is documented very widely and in detail and is in accordance with ISO/IEC 27000 series information security standards and is annually updated.

ISKE implementation contains the next steps:

1. Person responsible for IT staff and ISKE coordinator checking information assets and specification according to instructions.
2. Database mapping. Determination of security classes.

3. Determination of security classes for other information assets.
4. Determination of security level for information assets security classes.
5. Analysis for zones with different security needs.
6. Identification of typical models.
7. Creating a list of security measures.
8. Creating a plan for security measures implementation
9. Implementation of security measures
10. Control of the actual security situation, implementation of additional measures if needed.
11. Configuration and change management.

Analyzing all data, what is brought above it is possible to bring some big problems of ISKE. First of all, ISKE is designed for large corporations, not for small IT projects. Secondly, it is compulsory for state and local organizations (not just a handbook). Thirdly, it is over 3000 pages long, so this can scare small organizations.

2.4.5. ISO/IEC 12207 Systems and software engineering

This standard presents forty three different software processes and system contexts. Describes the whole testing process and opportunities for improvement of this process. It is important to notice that standard do not describe specific software development life cycle or method. The standard is suitable for a wide variety of the life cycle models, including agile.

Standard divides software lifecycle processes into three groups:

- Primary processes (5): Procurement – Delivery – Development – Exploitation – Maintenance
- Utility processes (8): Documentation – Configuration Management – Quality Assurance – Verification – Validation – Review – Auditing – Solving problems
- Organizational processes (4): Management – Infrastructure – Improvement - Training

Every process contains actions, actions contain works. Together standard provides 23 processes, and there are 95 connected actions. For example development process contains such actions as: system requirements analysis, system architecture design, software requirements analysis, software architecture design, software programming and testing, software integration, software qualification testing.

Important to notice that this standard do not describe the order of processes and actions and do not describe any of software lifecycle models.

2.5. Choosing the best standard for software component

Considering analyzed standards and methodologies we made following conclusions for every type of standard. Basing on this data, standards will be implemented in abstract information system in the next part. Standards types in the table below are separated by color and first one of every type suits better for small information system.

Table 8. Analysis of Standards

Quality/Standard	Understandable	Scope	Availability	Structure	Type
IEEE Std 830	Very simple, even for students and for people with small experience in IT.	37 pages	117 $ (IEEE)	Well-structured, logically divided into parts.	Requirements
ISO/IEC 27001	Simple and understandable.	34 pages	16 EUR (EVS) EST, ENG	Well-structured	Requirements
ISO/IEC 25010	Simple, overview is not detailed. Cursory review of different requirements.	33 pages	15 EUR (EVS) EST, ENG	Good structure	Requirements
RiskIT	Gives a great explanation of risk sources. Contains a lot of pictures and	107 pages	Free (ISACA)	Well-structured. In the beginning is brought a big scheme,	Risk Management

	schemes, what simplifies the documentation.			which demonstrates the structure of documentation.	
ISO/IEC 27005	Easy to understand, but difficult to implement to small information system. Too many aspects to analyze.	68 pages	140 EUR (EVS) ENG	Well-structured	Risk Management
FMEA	Simple and easy to follow. Despite the small size, can take quite much time to implement.	10 pages	Free (ASQ)	Simple step-by-step structure	Risk Management
ISO 5127	Covers many fields. Not difficult to find needed definition.	253 pages	31 EUR (EVS) EST, ENG	Well-structured, despite the large amount of information.	Documentation
ISO/IEC 26511-26515	Simple and understandable, contains five parts, which are logically divided.	141x5 pages	120x5 EUR (ISO)	Well-structured. Consists of five parts.	Documentation
EVS 8:2008	Easy to follow, very understandable. Reasonable to implement in Estonian projects.	112 pages	37 EUR (EVS) EST, ENG	Very well-structured and not a problem to find needed information.	Documentation
ISO/IEC 9294	Covers all stages of software development, but too cursory. Better if it would be one development component, but	14 pages	10 EUR (EVS) EST, ENG	Well-structured	Documentation

	in detail.				
ISO/IEC/IEEE 29119	Simple, helps to understand many testing techniques. Can be very useful in small information system.	240 pages	140x3 EUR (ISO)	Well-structured. There are three parts of this standard and two are under development.	Testing and Privacy
ISO/IEC 12207	Cursory and understandable information. Gives basics of testing process.	123 pages	25 EUR (EVS) EST, ENG	Well-structured	Testing and Privacy
ANSI/IEE Std 829	Too detailed, difficult to find needed information.	148 pages	Free (Internet)	Probably it should be divided into parts. Too many different parts and subparts.	Testing and Privacy
ISO/IEC 29101:2013	Contains serious information. Pretty difficult to understand and not necessary for simple information system. Contains difficult security schemes with different layers.	45 pages	125 EUR (EVS)	Well-structured	Testing and Privacy
ISKE	Unreasonable to implement in small IS, better suits for big government systems. Easy to follow and understand.	3800 pages	Free (RIA) EST	Despite the amount of data, standard is well-structured.	Testing and Privacy

3. Quality Assessment and Assurance Models

Through years more and more software is produced. The quality of software architecture however has an important role in systems exploitation, as it determines the maintainability and extensibility of a system. Modifications are harder to implement in systems, which architecture is a substandard condition [10].

Quality measurement is needed to achieve more efficiency in the project functions and to improve the delivery of services. The question to be answered in this part is how to assess the quality of e-services, how to check compliance with standards and provide possible solutions for project improvement using efficiency of standards.

Quality assessment is an opportunity to prevent failures and mistakes in products and destroying troubles when delivering product to customers. Quality assessment is connected to administrative activities and quality system so that requirements and aims for a product will be done. Quality can be defined by product users, clients and customers.

Another thing is quality assurance. This is not just testing, technologies and analysis. For some companies it is just boring, difficult and tedious procedure, but quality assurance is nonetheless essential. Quality can be achieved by using the standards. Provided in this work model is not the only possible and maybe is not the most efficient, therefore it needs to be tested.

Nowadays, there are a lot of opportunities to create a quality project, which will meet the specific standards. Every part of project can be made by using rules and regulations (Figure 7) which already exist and have been successfully used by many companies.

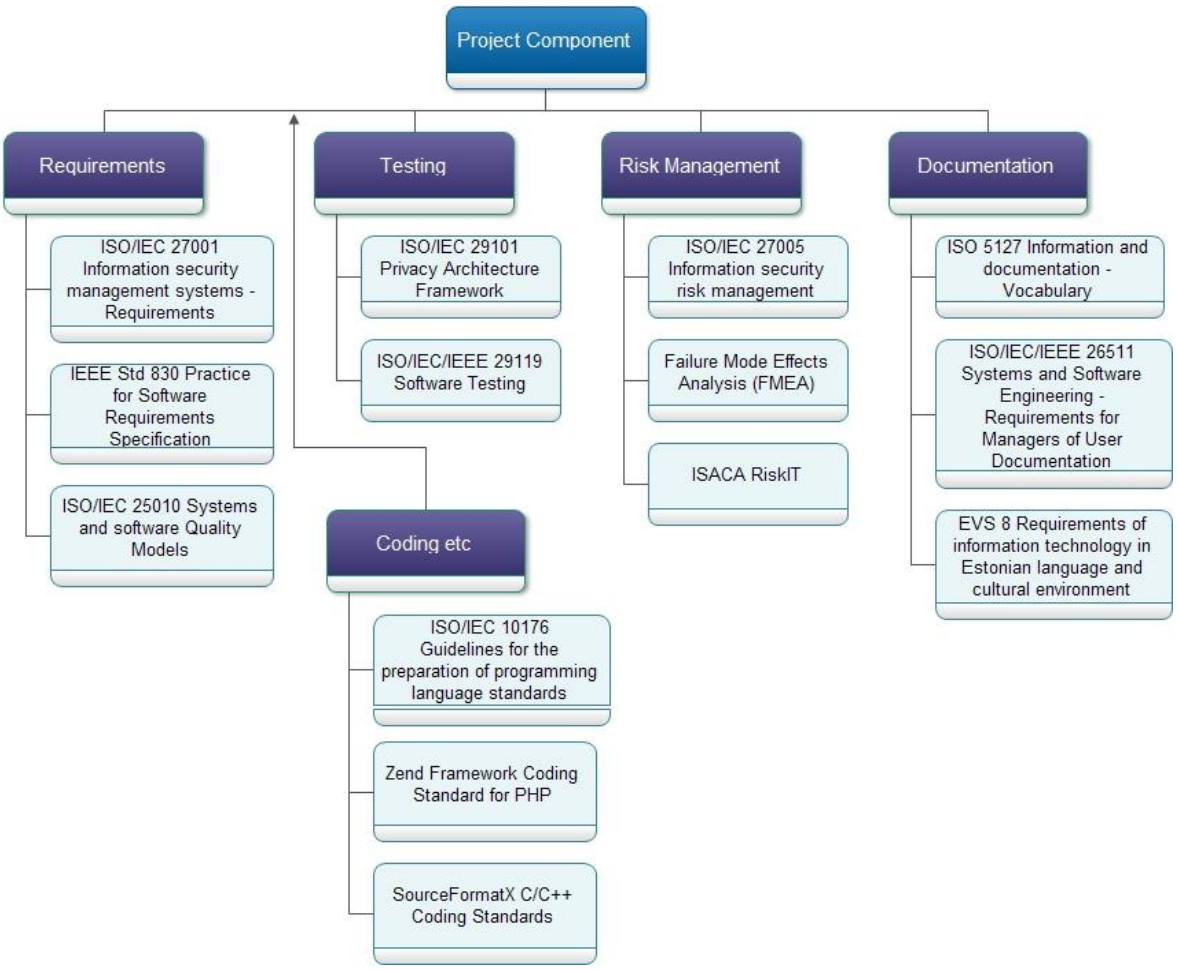

Figure 7. Project Components and Standards

Shown in the figure above components are very important for IT projects. There can be added other processes (e.g., usability, resource management, work environment). Listing of processes is not an aim of this work.

On the other side, in some IT projects standards can be used not fully or not used at all. In order to ensure in software quality, can be carried out research by the company as well as by client who is going to use the service. Research can be carried out using quality assessment and assurance models (Figure 8). In this work for analyze will be used CMMI model. This model was chosen due to many reasons:

- Important benefit is that chosen model supports Agile Software Development. In addition, CMMI is aimed to avoid chaos and risks, by giving great structure and detailed documentation to IT projects.
- This model is pretty young, first version appeared in 2002 (ver. 1.1), the latest in 2010 (ver. 1.3)
- There is no doubt, that for now in Estonia and in whole Europe are very popular different international ISO quality standards (ISO 9000, 9001, 9004), and CMMI is used mostly in the USA and in pretty large projects (Boeing or NASA). However, more and more companies, which want to create a strong and reliable foundation, choose CMMI (Tieto Software Technologies, IBM, HP, CGI) [11].
- Model was developed to increase the quality of the software. It is based on different analyses and the main aim of the model is to help finish project precisely in time, within budget and with good quality.
- This is not a standard with steps which needs to be certainly completed; CMMI is a collection of recommendations for improving processes within the organization. This means, that for passing a CMMI certification not all its points must be implemented, in such way this model can be implemented in small and medium projects.
- Model provided a big interest, because it is used by many United States Department of Defense (also known as Pentagon) and other U.S. Government contracts.
- Moreover, complementing the previous point, it can be said that using a label "rated at CMMI level" can be a great marketing move.
- Another point is that there is a lot of works about implementation different ISO standards, but not too many researches were made about implementation the CMMI model.

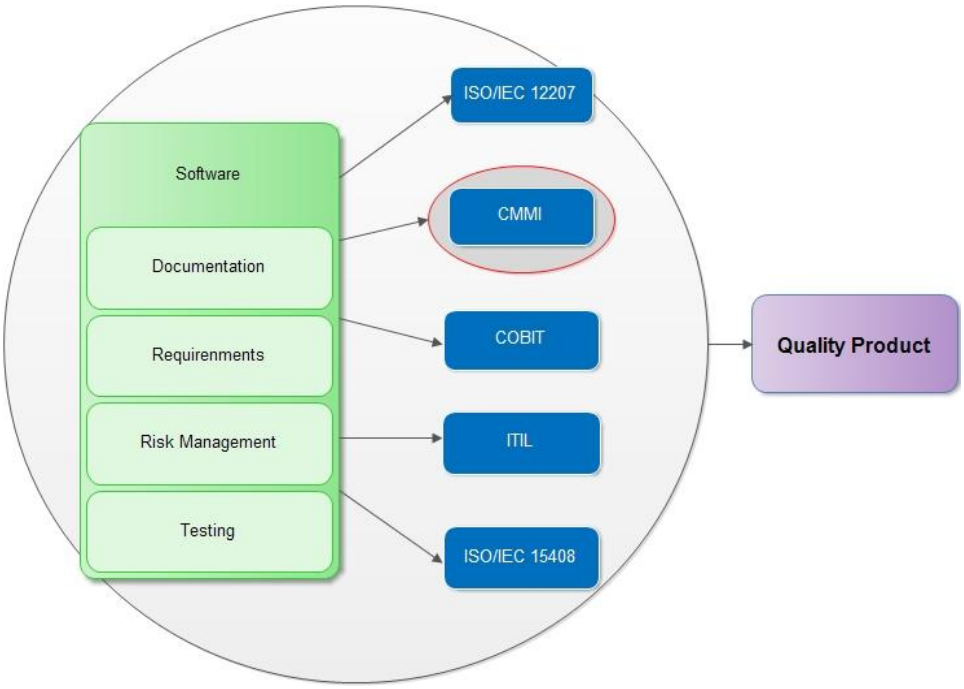

Figure 8. Quality assessment and Assurance Models

To find out, which quality evaluation method is better, we created a table and pointed out main characteristics, what are important for project developers, for clients and for company. Results show, that three of analyzed models are pretty similar, CMMI and ITIL are more detailed than ISO 9001.

Table 9. Comparison of Evaluation methods

	CMMI	ISO 9001	ITIL
Content	Best practices for software systems (derived from the IT industry). Divided into different levels, each level progresses to higher standard.	Main project processes are described superficially. (26 pages)	Best practices for IT service management and infrastructure. Collection of books, each for a specific practice within the IT service management.
Main focus	System and software engineering (e.g.,	Customer focus	IT service management (e.g., hardware life cycle)

	security, planning configuration)		
Main features (expectations)	Requirements, design, decision analysis, integrates project management, risk management	Quality policy. Resource management. Measure customer satisfaction	Organizational objectives, customer needs, service strategy, business objectives, release management etc.
Range of customers	U.S., China, Germany, Italy, India (mostly USA)	International (wide range)	England, some European countries, USA
Agile support (in documentation)	Yes (from ver. 1.3)	No (integratable)	No (integratable)

3.1. Capability Maturity Model Integration

CMMI contains different models (procurement, development, services). It extends and combines such models as – the Capability Maturity Model for Software, the System Engineering Capability Model and the Integrated Product Development Capability Maturity Model. Provides software procurement, development, maintenance and other processes best practices. In addition, model provides self-evaluation and comparison with others. Evaluation can be done using five level system:

1) The Initial Level (Level 1). Not many processes are defined, success of the whole project depends on individual effort. Process is unpredictable (schedules, budgets, functionality, quality) because it is constantly changing and modifying during the work progress.
2) The Repeatable Level (Level 2). Basic processes are created to track cost, schedule, and functionality. Project processes are based on experience with similar projects.

3) The Defined Level (Level 3). The process for management and engineering is documented, standardized, and integrated to a standard software process.
4) The Managed Level (Level 4). Software process and products are understood and controlled.
5) The Optimizing Level (Level 5). Continuous process improvement is enabled by process feedback and from new ideas and techniques [12].

The main opportunities and positive sides of CMMI are: opportunity to choose suitable model, detailed, designed specially for software developing organizations, focusing on continuous development (not only certification), can be used for certification and for self-evaluation and is for free on the Internet.

3.2. CMMI in Agile

Agile approach is very popular in software development in the last decade, therefore many of software development companies have their own agile methodology or use already created techniques. CMMI is not too popular in agile, because agile support is available and well described only from 2010 (ver. 1.3). Agile principles and practices are very beneficial to many organizations. They are team oriented. However, CMMI provides an "organizational-level" infrastructure and techniques to:

- Preserve information and knowledge over time.
- Structure and provide criteria for decision-making.
- Strengthen and normalize risk management.
- Methodically apply technical approaches.
- Focus on process improvements.
- Includes development best practices, but assumes that users know the basics.

In the table below are shown examples, how some of the CMMI practices can be mapped to agile methodology Scrum.

Table 10. CMMI and Scrum mapping

ID	CMMI Practice	Scrum
1	Develop an understanding with the requirements providers on the meaning of the requirements.	Review of Product Backlog with product owner and team.
2	Obtain commitment to the requirements from the project participants.	Release planning and Sprint planning sessions.
3	Manage changes to the requirements as they evolve during the project.	Add requirements changes to the Product Backlog. Manage changes in the next Spring planning meeting.
4	Identify mismatches between the project plans and work products and the requirements.	Daily standup meeting to identify issues. Release planning and Sprint planning sessions to address mismatches. Sprint burndown chart that tracks effort remaining. Release burndown chart that tracks story points that have been completed.
5	Establish a top-level work breakdown structure to estimate the scope of the project.	The standard tasks used in Scrum process combined with Scrum Backlog.
6	Define the project life-cycle phases upon which to scope the planning effort.	The Scrum process.
7	Estimate the project effort and cost for the work products and tasks based on estimation rationale.	Scrum Ideal Time estimate.

In the table above are shown some of the requirements management and project planning practices. In the same way can be mapped other practices like project monitoring, project control, quality assurance and others. Integrated CMMI into agile process is demonstrated in Figure 9.

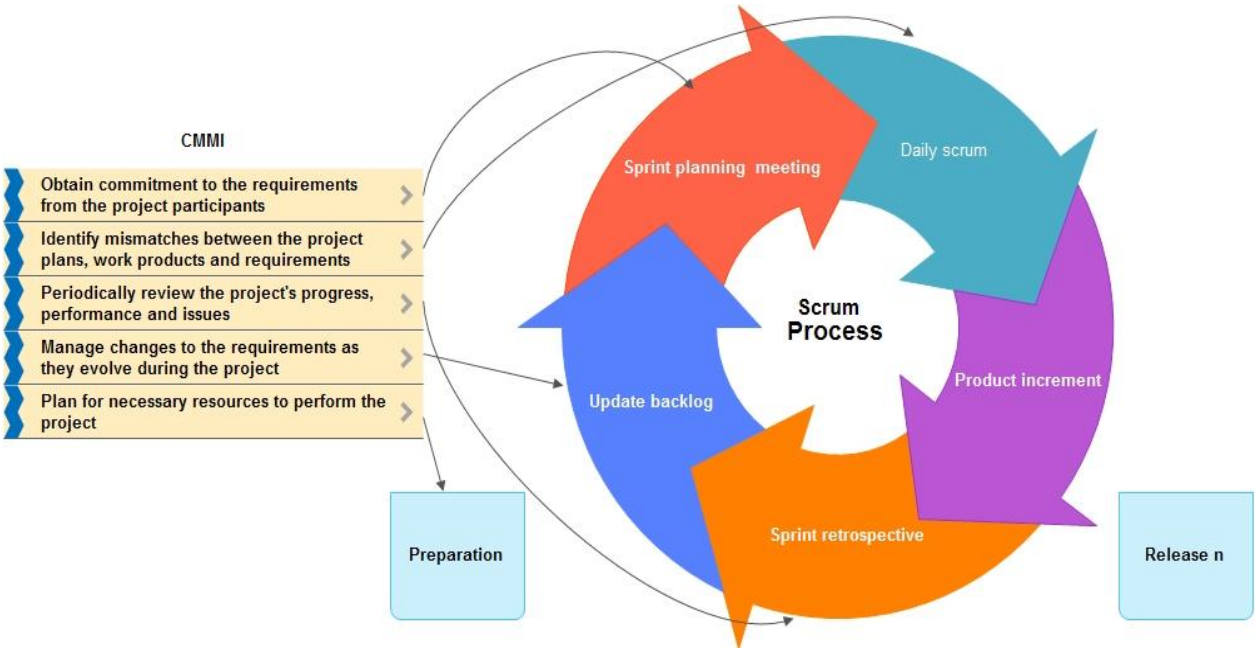

Figure 9.CMMI Integration into Agile

3.3. Information system for analyze

For analyzing compliance with standards will be chosen abstract system named Example. Let's assume that project was made without using any standards. However, system works without any noticeable errors. The system deals with selling tickets for intercity bus service. The goal is to show that standards can help find errors on earlier stages of project development or to prevent them, because the earlier the mistake will be found, the less cost will go to its elimination. Moreover, even small and simple information system can depend on many aspects and components (Figure 9), which are necessary for successful project.

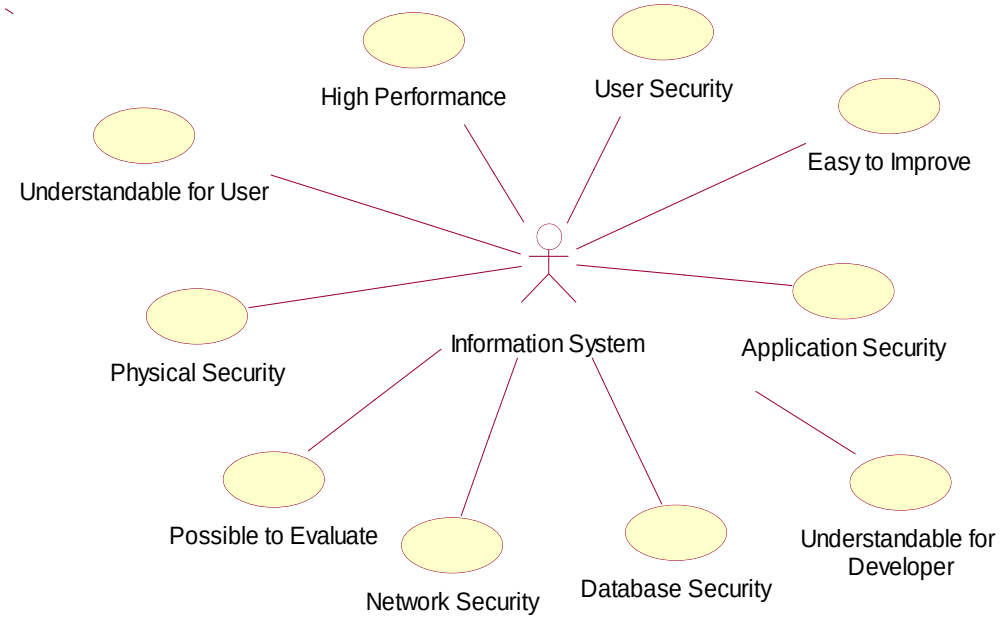

Figure 10. Example Information System Components

3.3.1. Measurement and Analysis Activities

CMMI has the following benefits for projects [13]:

- Helps to collect metrics of data across the organization
- Helps to control project performance
- Helps to establish organizational set of metrics goal
- Helps in transfer of learning

3.3.2. Measurement Objectives

First of all it is necessary to define the goals for measurement. To define, what steps of project are going to be measured. The measurement set should focus on customer objectives, process and project objectives which are critical for performance. Objectives to measure are components of project development. Every step of development will be measured for compliance the standards.

- Documentation
- Requirements
- Risks
- Testing and Security
- Usability

3.3.3. Specifying Measures

If some of the project parts will not meet the requirements of the model, when will be offered measures for elimination the problems.

- Provide a standard, which will help to reduce possibility of mistake
- Calculate the cost of the quality
- Calculate possible schedule variance

3.4. Monitoring of project risks

Every project, even small have some risks and problems. Even after testing developers cannot be convinced, that system will work without failures, especially that risk is not always testable. It can be said that IT risk always exists, whether or not it is detected or recognized by an enterprise.

For Example system will be used “ISACA RiskIT” framework which helps to identify and manage different types of risks in IT projects. RiskIT analyze is in part 2.2.2.

Here are possible risks that can affect the system:

Table 11. RiskIT Requirements

Risk	Risk Example	Affect	Risk Source
Popular trips functionality do not work as it should	Auto input do not work as it should	low	Uncertain requirements
Trip viewing	User cannot find	high	Bad design

function do not work as it should	detailed information for needed trip		
The system does not remember the date of the selected route	User can buy a ticket for the wrong date	high	Inadequate staffing
If there is an error, then system do not notify an error message	The user has no idea about the cause of the error, which can lead to the situation where user does not know how to work with the system	high	Bad design
Absence of help, what could be useful for user	The user has no idea about the cause of the problem , may lead to the situation where the user does not know how to use the services	high	Bad design
Changing the language of the system, causes the loss of page where the user was	Can be the reason of user not loyalty to the system	low	Bad design, inadequate staffing
Departure and destination swapping do not work	Manual input is not convenient for the user	low	Inadequate staffing and skills
Absence of e-mail, while buying a ticket	Loss of ticket	high	Bad design
After pressing a button system should answer in one second	The user can start to get nervous	high	Unavailable technology, inadequate staffing
The content of the system is only in one	Losing customers, who speak in another	middle	Uncertain requirements

language	languages		
Website keywords are not associated with the sale of tickets or the lack of content	Users do not understand the purpose of the system	middle	Bad design
HTML and CSS errors and warnings	Internal changes of the system may bring problems with page appearance	middle	Inadequate staffing
Web page displays differently in different browsers	The differences in design can complicate using of system	middle	Bad design, inadequate staffing
Absence of system mobile version	Impossible to buy a ticket from smartphone	middle	Unavailable technology, Bad planning

From CMMI model, the source of the most risks comes from the system, developers carelessness or from the lack of documentation and uncertain requirements. Some of the typical internal and external risk sources include the following:

- Uncertain requirements
- Not implementable design
- Unavailable technology
- Unrealistic schedule
- Inadequate staffing and skills
- Cost or funding issues

The standard can be like a mentor even for experienced developers, helping them to prevent risks on earlier stages of the projects. Another consequence from this part is that CMMI model can be used for project improvement as well, not only quality measurement. However analyzing the IEEE Std 830 and CMMI model requirement

specifications can be said that they are very similar and CMMI helps to verify if standard was well implemented by duplicating its steps.

3.5. Product Design Implementation and Documentation

The design implementation usually includes unit testing of the product components before sending them to product integration and development of end-user documentation. ISO/IEC 29119 covers most of the testing processes and parts of this standard can be used in our information system. Standard was analyzed in part 2.4.2.

For instance, in our Example information system we have a requirement that first name should not contain nothing else, but small or/and capital letters. This requirement is testable and can be tested using Selenium or other software. Considering ISO standard, after testing it is necessary to create well documented report. Below is shown an example of TEST-INCIDENT REPORT for our information system considering standard ISO/IEC 29119-1: Test Documentation.

Test-Incident Report:

- **Test-incident-report identifier:** TEST-1
- **Summary:** System allows registration with wrong input
- **Incident description:**

Table 12. Incident description

Test_ID	First Name	Last Name	E-mail	Password	Repeat Password
Id_01	Mihhail2	Marenkov	mihail@gmail.com	qwerty	qwerty

- **Expected result:** error message
- **Actual result:** page with user data
- **Anomalies:** result using this input does not answer requirements
- **Date and time:** 29.01.2014 20:14

- **Attempts to Repeat:** were used other first names with numbers
- **Testers:** Mihhail Marenkov

Analyzing this standard, we can make a consequence that using of all standard steps and techniques can be necessary only for big and serious projects, for small information systems there is no need in using all of the standard benefits.

CMMI model helps to verify if all system components were tested as appropriate. Model provides examples of unit testing methods (e.g., statement, branch, predicate, path, boundary, functional testing). In addition, it provides a verification process which helps to ensure that selected work products meet their specified requirements.
CMMI Verification process provides different verification methods to verify adequacy of system design. For example:

- Path coverage testing
- Load, stress, and performance testing
- Decision table based testing
- Functional decomposition based testing
- Test case reuse
- Acceptance testing

Selection of those methods should begin with the definition of system requirements to ensure that they are testable. This risk can be solved or minimized by using standards in every part of project development.

3.6. Requirements Development and Management

Considering standard "IEEE Std 830" there should be a small documentation before writing a requirements with introduction and overview, but I will analyze only a part of standard which is connected with different types of requirements.

There are listed different types of requirements for the system. Considering the standard requirements should cover the system areas, which are listed in the part 2.1.1.

Function: Possibility to watch more information about the trip
ID: FR01
Purpose: System must show to user information about bus stops, trip date, duration, departure days, bus lines, carrier, line type, arrival and departure time
Actors: user, system
Input: User is watching information about the trip
Example scenario:

Subject	System
User presses the button "View" to watch information about trip he is interested in	Displays information about the trip

Output: User successfully got all needed information
Priority: 3

Function: Possibility to search for a trip by entering arrival and departure stops
ID: FR02
Purpose: System must show to user all trips which will lead to the point of destination, this will make search for user simpler
Actors: user, system
Input: User is searching a trip
Example scenario:

Subject	System
User enters arrival and departure destinations and presses a "Search" button	Displays all possible trips with needed destination

Output: User successfully found needed trip
Priority: 3

Function: Possibility to stay on same page, if the page language was changed
ID: FR03
Purpose: System must display the same page, if the system language was changed
Actors: user, system
Input: User is changing a language
Example scenario:

Subject	System
User changes a language from Estonian to English	Displays the same page translated into English

Output: User stays on same page and do not need to search this page again
Priority: 2

Function: Possibility to receive a failure message if user input is not correct
ID: FR04

Purpose: System must help user to find an error and to fix it as fast as possible
Actors: user, system
Input: User is writing an arrival city with a mistake
Example scenario:

Subject	System
User inputs city name with an error. (e.g., Talinn instead of Tallinn)	Displays an error message and automatically changes a mistake

Output: User have found needed trip despite an error
Priority: 2

Function: Possibility to get the information from the system on different languages (Estonian, Russian, English)
ID: NFR01
Purpose: System must be on language, which is understandable for user
Actors: user, system
Input: User do not understand the system language
Example scenario:

Subject	System
User do not understand the language of the system and chooses a needed language	Correctly translates the system to needed language

Output: User chooses needed language
Priority: 3

Function: Moving between system links should take less than one second
ID: NFR02
Purpose: System must be as fast as possible, this will save users loyalty
Actors: user, system
Input: User presses a button to go to another system link
Example scenario:

Subject	System
User goes to another link	System answers faster than in one second

Output: User got needed result less than in one second
Priority: 3

Such approach helps to exclude the possibility of non-testable requirements. Considering this it is possible to make a consequence that standards should be used on every stage of system development, because absence of standards on one stage can make them worthless or hardly implementable on another.

To evaluate the quality of project requirements CMMI model provides a management process (Figure 10) what helps to manage all requirements generated by the project, technical and nontechnical as well as requirements which are provided by organization.

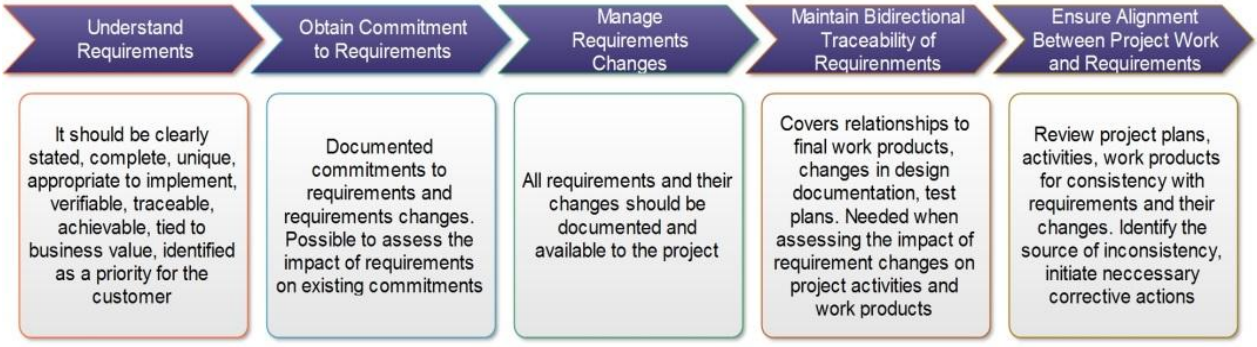

Figure 11. CMMI requirements management process

Requirements which are made using standards and supplemented using CMMI model can bring a lot of benefits to every company. Of course, such approach requires additional work to be done in developing of even small information system. However, with comparatively low investments, the functionality of project can increase considerably.

3.7. Implementing CMMI in Real System

In order to make sure that proper use of standardization can help to eliminate problems and improve project quality we have chosen "Tallinn University of Technology" Study Information System (ÕIS). University electronic student information system is used to ensure better access to educational information. There are a lot of opportunities for students in system (e.g., academic calendar, view subjects' data, view the results of studies, and submit applications to apply for scholarship). According to the statistics, there are nearly 13500 students and 1114 teachers [14]. Consequently, system is constantly overloaded, has a wide range of activities and consists of many processes. Chosen system is not an ordinary project, therefor, for quality evaluation was chosen a

specific model. The figure below shows a selection of IT standards and methodologies in two-dimensional basis. On the vertical axis is shown the scope of the area – from wide to specific. On the horizontal axis is the level of methodology detail in number of pages, from overview to detail.

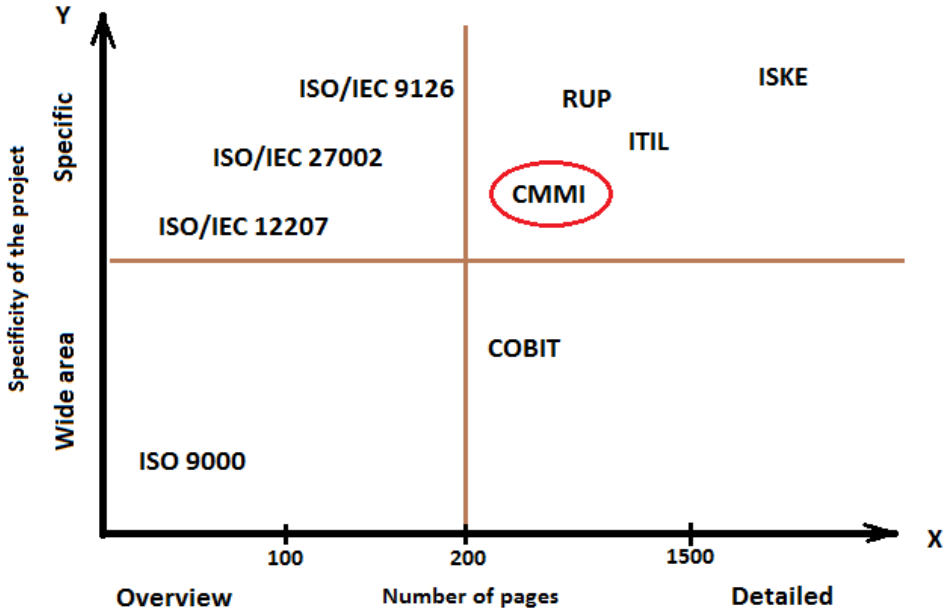

Figure 12. Selection of Standards and Methodologies

In this part we will evaluate a usability of Study Information System. Decision to investigate the usability topic and perform these tests wasn't just casual. Since the very beginning of using Studies Information System, there was a feeling that it needs to be more user-friendly, as performing some specific actions was embarrassing, very time consuming and on frequent occasions led to confusion. The activity diagram below demonstrates the steps of solution that we propose to evaluate and improve processes in projects. Of course, such solution is not the only possible and should be carefully tested. Such approach can be applied to every process (e.g., risk management, planning). Each IT project process is unique and can be represented as follows:

$$(K1,\ldots,Kn) \in P, \text{ where}$$

- P – IT project
- K1,..., Kn – project processes.

Every process can be represented as a set of parameters: (r1,…, rn) ∈ Kn. Standard S describes the best practices of project process, their structure, parameters and other attributes.

$$(S1,\ldots, Sn) \in ((r1,\ldots, rn) \cup (K1,\ldots,Kn)) \in P$$

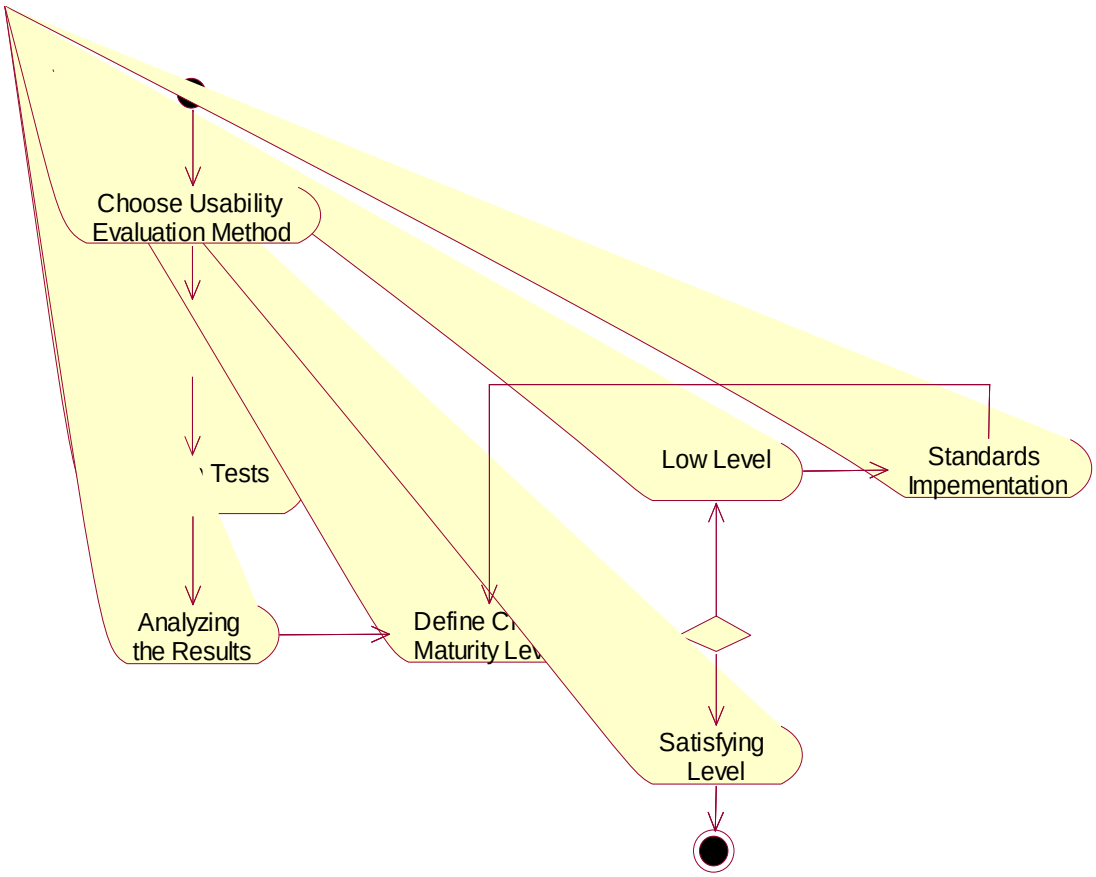

Figure 13. Usability Evaluation Activity Diagram

3.7.1. Choosing Usability Evaluation Method

Usability analyzes was made using "Think Aloud Protocol" independently from our research. For tests there were chosen 5 persons, they performed testing scenarios and author has pointed out main usability errors which were encountered by users [15]. Our usability analysis is based on described above consequences and tests. As a solution, to avoid possible usability errors, we provide to evaluate IT project quality using CMMI model and if needed improve quality using IT standards.

3.7.2. Analyzing Test Results and Defining CMMI Level

Considering CMMI model, usability has a significant influence on the architecture. The main points on which we should pay attention are: duration of performing a test, simplicity and operability of the task performance, proper use of the navigation interface, users' feedback.

Since evaluation method is chosen and tasks for users to evaluate usability are created, we can start to perform and analyze test results. After performing tests and analyzing the results we need to define the maturity level of the process. CMMI levels were describes in part 1.1. When the process level is defined, we can provide analysis, how could described issues change if they were made in compliance with standards? Most of issues are connected to usability of the system therefore for analyze we use standard ISO 9241 which provides requirements and recommendations relating to the attributes of the hardware, software and environment that contribute to usability, and the ergonomic principles underlying them [16].

1. *You want to create a new unique password for TUT systems. Please explore the navigation interface and find this option.*

   From analysis only one person has found a proper menu option from the first attempt. The most logical location of this option in respondents' opinion was My data section which turned to be a wrong assumption instead. Even though the proper interface has been still found by all the users in timely manner, there were many drawbacks and misusing the interface in this case. Considering the analysis and failed tests, process can be marked as initial, because it is unpredictable and poorly controlled. ISO 9241-12 "Presentation of Information" recommends that understandable phrases should be used to improve user's

control [17]. For example, simply “Change Password” for changing a password. In addition, if tasks have specific steps, when dialogue help should be provided.

2. You have never got any notification e-mails from the system. Please check your subscription status to university newsletters.

   Many users successfully dealt with this task, however some of them had issues with menu navigation. Nevertheless, two users clicked on the name in the upper right corner first, which resulted in new pop-up window with user basic information. Then user has clicked on the magnifier icon by the name and got the same pop-up window again. Even though this window represents user’s data, this wasn’t proper interface to fulfill current task. This made user close the pop-up window again and look for the different option to get to user profile details. The overall experience with this task is positive, but there still was 1 user who has been confused, and as a result, spent extra time and efforts performing the simple task. This occasion refers to a fact, that this pattern should be more intuitive. Considering the analysis and failed tests, process can be marked as initial. ISO 9241-14 provides recommendations for the design of menus used in user-computers dialogues [18]. Standard recommends that menu headings should be short. For example “Subscribe for e-mail notifications” can be changed to “Notifications”.

3. You need to get an additional information on the course called "Testing of Digital Systems". Find dedicated web page for this course. The course code is IAF0050.

   Users have not managed to find the proper interface in timely manner. According to user’s feedback from analyze, there might be more intuitive and user-friendly terminology used. Considering the analysis and failed tests, process can be marked as initial. Standard ISO 9241-110 “Dialogue principles” (2006), gives clear guidance about how should dialogue system react in case if user needs some help [19]. System should prevent giving information to user what is not connected with the successful finishing of the task. Standard ISO 9241-13 “User guidance” provides recommendations for the design and

evaluation of guidance attributes of systems including help, feedback and error management [20]. Standard says that appropriate time for help should be provided and it should not be intrusive. However, now "Help" button is somewhere on top and help for creating a study plan "Abi" is not detectable from the first sight and in fact is not translated into English.

4. You need to submit the study plan for forthcoming semester. Check if there are relevant instructions provided on how to perform this action.

   All 5 participants had trouble performing this task. According to current system implementation, the correct way would be navigating to Help section first and then finding the proper menu section or to go navigate to proper section first and then find the "Abi" button, which is not detectable from the first sight and in fact is not translated into English either. One way or another, but according to users' behavior the pattern currently implemented in ÕIS was not so obvious to follow. Among the first things users have checked were Study plans and Create Timetable menu options, where according to users' feedback they expected to get see clear instructions required in order to get the study plan submitted. Considering the analysis and failed tests, process can be marked as initial.

5. Examination period has been started. Please register yourself to the exam, relevant course code is IAF0050.

   Considering the performed analysis the average time it took the users to register to an exam is 1:47 minutes which is considerably high result, meaning that the average user should be definitely able to perform this action in shorter time period and with less effort. Considering the analysis and failed tests, process can be marked as initial. This problem could not occur if while project designing were used standards ISO 9241-2010 "Human-centered design for interactive systems" principle that project can be improved by its evaluation by users [21].

3.7.3. Obtaining a Satisfactory Level

After the implementation of standards, all described above usability weaknesses must be eliminated and evaluated at least as Level 3, which means that process is documented, standardized, and integrated to a software process.

Other possible usability improvements:

- Larger font size, considering W3C Web Designing standards [22].
- Button names along with most of the menu options should be renamed according to the actions they perform from user's perspective. ISO 9241-14 "Menu Dialogues" can be used.
- English translations should be added to all the menu items and texts. For this can be used EVS-EN 15038 "Translation services – Service requirements" standard. Standard describes needed professional competences of translators, competences of reviewers and procedures in translation services [23].
- Additional features for usability improvement from ISO 9241-151 "Guidance on World Wide Web user interfaces". For example, can be added small pop-up window that appears when the mouse pointer is moved over an interaction object and that shows explanatory text or help information. Also, standard advices to use site map with textual or graphical overview of the complete navigation structure of a website [22].

Usability and usability evaluation are the areas of intense interest among web site designers and developers. The main features of usability and its principles are still in the investigation stages, which makes the research on this topic a very important. In addition, we think that by solving at least those usability problems (our analysis was not very deep and detailed) "ÕIS" can become more user-friendly, efficient and will make life of students much better. Of course, standards are not free, but efficiency from using them is obvious. Unfortunately, now it is very difficult to change the usability of analyzed system, but if system would have been developed using provided model, there is a huge possibility that errors would have not appeared, therefore it is advised to fight with errors and implement standards on the earlier stages of the project.

4. Summary of practical assessment results

Based on the analysis of the implementation of standards and on the quality model implementation could be pointed out main advantages and disadvantages of using standardization in IT projects.

4.1. Advantages of using standards

In this thesis were described a lot of benefits of standards. Here are represented main advantages of standardization and quality assurance using CMMI model, which were clearly noticed during the analysis and implementation in projects:

1. The standardizing approach and the inclusion of standardization support in the system development offer not only global recognition and opportunity to compete on a larger number of markets around the world; moreover, standards additionally offer opportunity to create a quality and reliable software. Of course, standards do not explain how to create an excellent product, but it provides all needed components and main requirements for creating a good quality product.

2. Since standardization support is a part of systems development and it is available for the entire life cycle of software, the offered standards and CMMI model makes it possible to test, develop and assess the software without stopping the development process.

3. Implementation of standards requires additional work to include needed functionality or documents. Quality assessment using CMMI model do not take a lot of time, if software follows any standards.

4. Standardization saves time by preventing possible failures and errors. This is important in large systems where minor error in the beginning of project may crash the whole system in the end. Continuous using of CMMI model helps to prevent such situation as well.

5. From security perspective standardization and continuous quality evaluation is very necessary, because unsafe system can bring problems to users by sharing their personal data as well as to system developers by local legislation for distribution private data.

6. During analysis we made a consequence that it is strongly recommended that if standards were used in one project stage, when developers should continue using standards in other stages. All stages are very connected; absence of standards in one stage can make them worthless on another.

7. The absence of standards or inability to use them can lead to worthless or duplicated work and can considerably reduce the quality of project.

4.2. Disadvantages of using standards

During the standards and quality model implementation were marked some inconveniences and disadvantages of that approach. The most important drawbacks are represented next:

1. On the other side standards can remove the creative side of project. If all web sites were made using same design standards it would be pretty boring.

2. Different standard certificates (for example ISO certificate) do not guarantee that the company delivers projects with a great quality. It just certifies the company. So it can mislead customer that certified company will definitely develop quality project.

3. Standards can be a problem for small organizations, because of money. Some of them are pretty expensive. In addition, there can be need for additional employee for quality management.

4. Standards implementation and quality assessment are fairly long processes. Can be difficult to implement in short time.

All those disadvantages are single cases, and they cannot occur in serious corporations which focused at creating a quality product.

4.3. Advice for using standardization in IT projects

1. Some versions of standards are withdrawn or revised. Therefore we recommend to check if there a newer version of standard before implementation. Otherwise, project will not comply with the latest technical or documentation requirements.

2. Different standards and quality assessment models are created for projects with different scale. For instance, analyzed in this work CMMI model is more suitable for larger projects and companies. In addition this model is pretty difficult.

3. Another important detail that we recommend to use is that standards, as well as quality models can be implemented partly. For example, analyzed standard ISO/IEC/IEEE 29119 Software Testing consists of five parts, but it is not necessary to use all of the standards features.

4. Creating projects using standards takes more time and efforts. Therefore, it is advised, that companies which use standards while creating an IT project should inform customers about used standards and about their benefits. So companies can have a loyal costumer and earn additional money for quality product.

4.4. Future work

Further research directions are related to implementation quality assessment model in real projects. We plan to expand the scope including different quality models and wider selection of IT standards. Can be made an analysis of different project parts using CMMI, SPICE, Common Criteria or another quality assessment model. Find out if there is any compliance with standards. Specification of CMMI model and basic benefits from standardization in abstract project is the step in this direction.

Like we analyzed ÕIS information system in this work, other projects can be evaluated in same way. Moreover, in future researches can be analyzed not only parts of usability, but also requirements, documentation and code structure of real projects.

Quality assurance - aiming to provide safe, reliable, documented and well-structured projects to system users and customers – is unavoidable goal which can be reached using available tools.

Conclusions

The aim of the thesis is to give an overview of standards and to use standardization to create quality product. The standards implementation is based on an intercity bus service information system (abstract project) and on real ÕIS information system. The overview shows the main standardization opportunities, benefits for IT projects and how they can assure quality in projects. The thesis presents the standards implementation and evaluation by using one of the quality assessment models. In this work for evaluation was chosen CMMI model. Afterwards, the analysis of implementation and evaluation are represented for every project stage. In addition, we describe positive and negative sides of standards implementation and giving recommendations.

The idea behind creating standards was to develop rules for creating a good and well-structured product, but not an excellent product. The absence of standards or inability to use them can lead to worthless or duplicated work that can considerably reduce the quality of project, increase development time, worsen security and lead to loss of data. Those are the main problems that standardization aims to solve.

Moreover, standardizing approach and the inclusion of international standardization support in the system development offers global recognition and opportunity to compete on a larger number of markets around the world.

After describing different standards and their benefits we implement them on an abstract project and define main positive and negative sides of them in the system life cycle development process.

Analyze based on implementing standards to a real and abstract information system points out main advantages of standardization that were clearly noticed during the implementation. One of the main advantages is ability to prevent errors from the

appearance. It makes it possible to provide a product with good quality to clients. The analysis points out that every issue, which was found in ÕIS system, could not occur if it would be developed in compliance with standards. Moreover, using standards in abstract project and afterwards measuring the quality of it with CMMI (quality assessment model) showed that project meets all model needs and requirements, which proves the high quality of the project.

Resümee

Antud töö eesmärgiks on anda ülevaade standarditest ning kasutada standardeid selleks, et luua kvaliteetne toode. Standardite rakendamine põhineb bussiliikluse infosüsteemi (abstraktne projekt) ning ÕIS (Tallinna Tehnikaülikooli Õppeinfosüsteem) infosüsteemi põhjal. Ülevaade näitab peamisi standardimise võimalusi, kasu IT-projektidele ja kuidas nad on võimelised tagama kvaliteedi projektides. Lõputöös on esitatud standardite rakendamine ja hindamine kvaliteedi hindamise mudeli abil. Selles töös valisime hindamiseks CMMI mudeli. Seejärel rakendamise analüüs ja hindamine on esitatud projekti iga etappi jaoks. Lisaks, kirjeldasime positiivsed ja negatiivsed küljed standardite rakendamisest ja andsime soovitusi.

Standardite loomise idee oli välja töötada reeglid, et luua hea ja hästi struktureeritud toode, kuid mitte suurepärane toode. Standardite puudumine või võimatus neid kasutada võib olla kasutu või dubleeritava töö põhjuseks, mis võib oluliselt alandada projekti kvaliteeti, suurendada arendamise aega, halvendada turvalisust ning olla andmete kadumise põhjuseks. Need on peamised probleemid, mida standardiseerimine püüab lahendada.

Seda enam, standardiseerimise kasutuselevõtmine ning rahvusvaheliste standartide toetuse kasutamine süsteemide arendamisel pakub maailma tunnustust ning võimalust konkureerida maailma suurematel IT turgudel.

Pärast erinevate standardite ning nende eeliste kirjeldamist me rakendasime neid abstraktses projektis ning määratlesime nende peamised positiivsed ja negatiivsed küljed süsteemi elutsükli arendamise protsessis.

Analüüs mille aluseks oli standardite rakendamine reaalses ning abstraktses infosüsteemides, märgib peamisi standardiseerimise eeliseid, mis olid selgelt määratud rakendamise ajal. Olulisem eelis oli võime vältida vigu. See muudab võimalikuks

kliendile toode pakkumist hea kvaliteediga. Analüüs viitab sellele, et iga probleem mis oli leitud ÕIS süsteemis võiks mitte esineda, kui see projekt oleks arendatud standardite järgi. Lisaks, kasutades standardeid abstraktses projektis ning hiljem projekti kvaliteedi hindamisel CMMI mudeli abil on näha, et projekt vastab kõigile mudeli vajadustele ja nõuetele, mis tõestab projekti kvaliteedi.

References

[1] M. Chemuturi. Mastering Software Quality Assurance: Best Practices, Tools and Techniques for Software Developers.J. Ross Publishing, 2010.

[2] M. Schmidt. Implementing the IEEE Engineering Standards. Sams Publishing, 2000.

[3] R. Strehlow. Standardization of Technical Terminology: Principles and Practices. American Society for Testing and Materials. Baltimore, July 1988.

[4] J. Tepandi. Software Quality and Standards [Internet]. Tallinn University of Technology, Institute of Informatics, 2013. Available from: http://deepthought.ttu.ee/users/tepandi/pdf/tks-loeng.pdf

[5] ISO Standards [Internet]. Available from: http://www.iso.org/iso/home/standards

[6] K. Jakobs. Standardization Research in Information Technology: New Perspectives. IGI Global, 2007.

[7] IEEE Recommended Practice for Software Requirements Specifications (Reaffirmed 1998). Institute of Electrical and Electronics Engineers (IEEE). IEEE Std 830, 3p.

[8] The Risk IT Framework [Internet]. ISACA, 2010. Available from: https://www.isaca.org

[9] The International Software Testing Standard [Internet]. Available from: http://www.softwaretestingstandard.org/

[10] J. Põld, T. Robal, A. Kalja. On Proving the Concept of an Ontology Aided Software Refactoring Tool. In: Databases and Information Systems VII, Selected Papers from the Tenth International Baltic Conference, (DB&IS 2012). Frontiers in Artificial Intelligence and Applications, IOS Press, 2013, 84-94.

[11] CMMI Published Appraisal Results [Internet]. CMMI Institute. Available from: https://sas.cmmiinstitute.com/pars/pars.aspx

[12] CMMI for Development Version 1.3 [Internet]. Software Engineering Institute. Available from: http://www.sei.cmu.edu/reports/10tr033.pdf

[13] CMMI Implementation Guide [Internet]. A Practitioner's Perspective. Available from: http://www.free-ebooks.net/ebook/CMMI-Implementation-Guide/pdf/view

[14] TUT Number of Students [Internet]. Tallinn University of Technology. Available from: http://www.ttu.ee/tugistruktuur/aastaraamat/oppetoo-32/uliopilaste-arv/

[15] M. Solman. Supervisor: V. Viies. Usability Concepts and Methods: TUT Students Information System Case Study. Bachelor's thesis, Tallinn University of Technology, Tallinn, Estonia, 2013.

[16] D. Wigdor, J. Jacko. Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications, Third Edition (Human Factors and Ergonomics). CRC Press, 2012.

[17] Ergonomic requirements for office work with visual display terminals – Part 12: Presentation of information. International Organization for Standardization (ISO). ISO 9241-12:1998.

[18] Ergonomic requirements for office work with visual display terminals – Part 14: Menu dialogues. International Organization for Standardization (ISO). ISO 9241-14:1997.

[19] Ergonomics of human-system interaction – Part 110: Dialogue principles. International Organization for Standardization (ISO). ISO 9241-110:2006.

[20] Ergonomic requirements for office work with visual display terminals – Part 13: User guidance. International Organization for Standardization (ISO). ISO 9241-13:1998.

[21] Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems. International Organization for Standardization (ISO). ISO 9241-210:2010.

[22] J. Zeldman, E. Marcotte. Designing with Web Standards.New Riders, 2009.

[23] Translation services – Services requirements. Estonian Centre for Standardisation (EVS). EVS-EN 15038:2007.

[24] Ergonomics of human-system interaction – Part 151: Guidance on World Wide Web interfaces. International Organization for Standardization (ISO). ISO 9241-151:2008.

Appendices

Appendix 1

Implementing Quality Assessment Model in IT Project

Mihhail MARENKOV[a,1], Taavi VALDLO[b], Vladimir VIIES[a]

[a] *Department of Computer Engineering, Tallinn University of Technology*

[b] *Department of State Information System, Ministry of Economic Affairs and Communications for Estonia*

Abstract. The goal of this research is to describe the best way of standards implementation in information system, by giving information about benefits of standards. The standards implementation is based on abstract project and on real information system. Research will show the main standardization opportunities, benefits for IT projects and how standards can assure quality in projects. In Estonia and in Europe there are a lot of electronic services. However, the quality of the services has not been measured in detail. Standards specify general principles and recommendations for initialization, implementation maintenance and improvement of information security management. Most of modern technologies are based on standards. They are working in the background and users notice the importance of them only when problem arise. The best way to measure the quality of project is using the quality assessment models. There are dozens of such models, which are provided by the world´s leading companies in development of audit standards. Several aspects of the model are discussed – how it works, how it can be used to measure the quality, provide the quality assurance and check the compliance with standards.

Keywords. IT standardization, quality assessment, quality assurance, quality models

Introduction

Every day IT companies from all over the world develop new projects and this process will never end. The quality of specification, design, code refactoring, software quality assurance becoming more and more important in competitive IT organizations. In recent time more emphasis is put to improve quality using already created tools and techniques for software development. Standardization is an activity to establish

[1] Corresponding Author: Mihhail Marenkov, Tallinn University of Technology, Akadeemia tee 15A, 12618 Tallinn, Estonia; E-mail: mihhail.marenkov@gmail.com

solutions to a repetitive situation. This is a tool for fixing requirements and improving quality, standards represent universal classes of demands. The choice of possible standards and methodologies is quite large. What to choose depends on the project needs.

Through years more and more software is produced. The quality of software architecture however has an important role in systems exploitation, as it determines the maintainability and extensibility of a system. Modifications are harder to implement in systems, which architecture is a substandard condition.

Some standards are very specific and detailed. For instance: ITIL (Information Technology Infrastructure Library), CMMI (Capability Maturity Model Integration), XP (Extreme Programming). Other can be comprehensive (not large), like ISACA, ISO/IEC 9126, 12207. All these frameworks can be divided into information technology oriented and general. Most popular general are: ISO 9000, Six Sigma.

The goal of standardization is ensuring the competitiveness and product quality, minimize expenses, improve the level of security for human life and health, ensure the rational use of resources, find optimal solution and many others. In addition to that, company what is using standards knows what to wait from IT, before the project starts and it knows what users are waiting from IT.

Quality measurement is needed to achieve more efficiency in the project functions and to improve the delivery of services. The question to be answered in this research is how to assess the quality of e-services, how to check compliance with standards and provide possible solutions for project improvement using efficiency of standards.

Quality assessment is an opportunity to prevent failures and mistakes in products and destroying troubles when delivering product to customers. Quality assessment is connected to administrative activities and quality system so that requirements and aims for a product will be done. Quality can be defined by product users, clients and customers.

Another thing is quality assurance. This is not just testing, technologies and analysis. For some companies it is just boring, difficult and tedious procedure, but quality assurance is nonetheless essential. Quality can be achieved by using the standards.

Quality Assessment and Assurance Models

Nowadays, there are a lot of opportunities to create a quality project, which will meet the specific standards. Every part of project can be made by using rules and regulations (Figure 1) which already exist and have been successfully used by many companies.

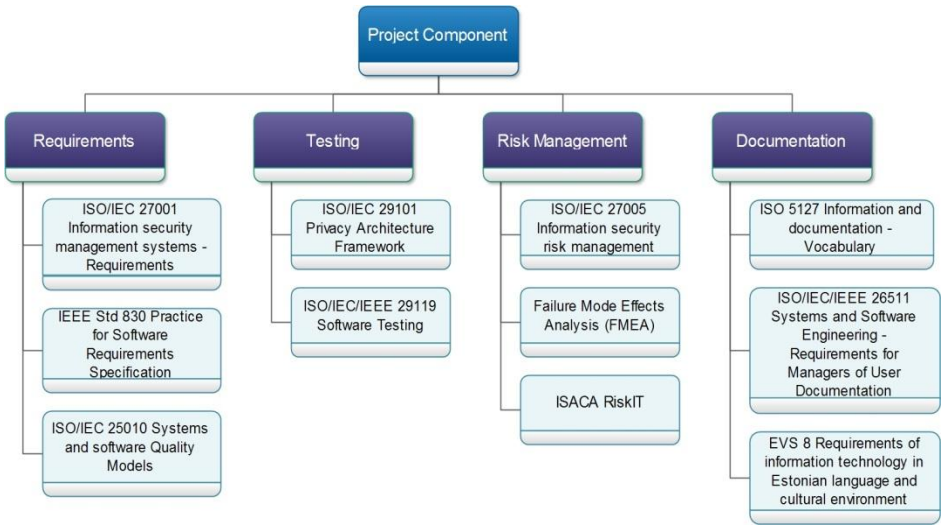

Figure 14. Project Components and Standards

On the other side, in some IT projects standards can be used not fully or not used at all. In order to ensure in software quality, can be carried out research by the company as well as by client who is going to use the service. Research can be carried out using quality assessment and assurance models (Figure 2). In this article for research will be used CMMI model.

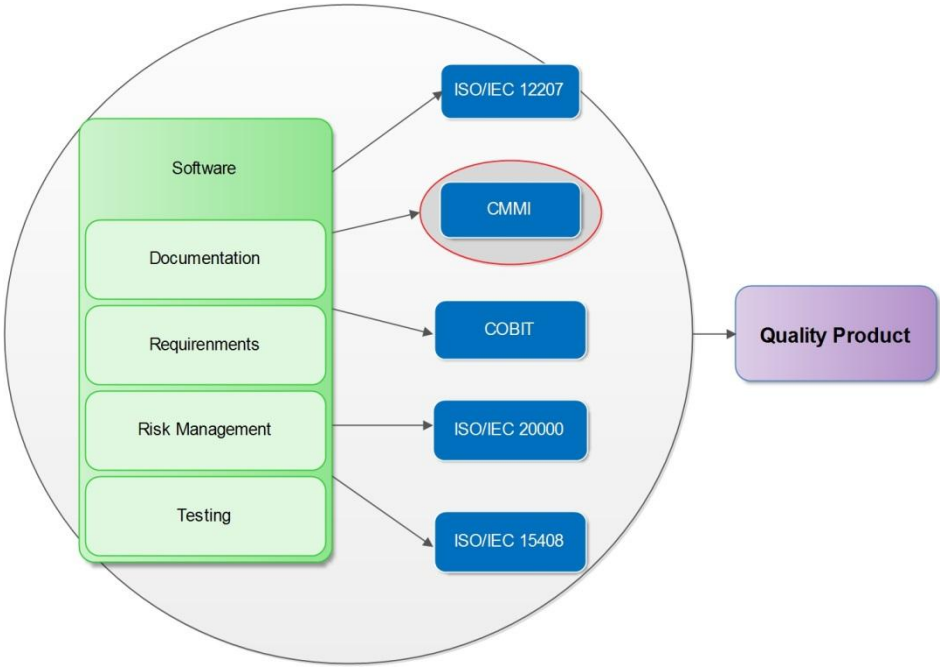

Figure 15. Quality Assessment and Assurance Models

Capability Maturity Model Integration

CMMI contains different models (procurement, development, services). It extends and combines such models as – the Capability Maturity Model for Software, the System Engineering Capability Model and

the Integrated Product Development Capability Maturity Model. Provides software procurement, development, maintenance and other processes best practices. In addition, model provides self-evaluation and comparison with others. Evaluation can be done using five level system:

- **The Initial Level** - not many processes are defined, success of the whole project depends on individual effort. Process is unpredictable (schedules, budgets, functionality, quality) because it is constantly changing and modifying during the work progress.
- **The Repeatable Level** - basic processes are created to track cost, schedule, and functionality. Project processes are based on experience with similar projects.
- **The Defined Level** - the process for management and engineering is documented, standardized, and integrated to a standard software process.
- **The Managed Level** - software process and products are understood and controlled.
- **The Optimizing Level** - continuous process improvement is enabled by process feedback and from new ideas and techniques.

The main opportunities and positive sides of CMMI are: opportunity to choose suitable model, detailed, designed specially for software developing organizations, focusing on continuous development (not only certification), can be used for certification and for self-evaluation and is for free on the Internet.

Information System for Analyze

For analyzing compliance with standards will be chosen abstract system named Example. Let's assume that project was made without using any standards. However, system works without any noticeable errors. The system deals with selling tickets for intercity bus service. The goal is to show that standards can help find errors on earlier stages of project development or to prevent them, because the earlier the mistake will be found, the less cost will go to its elimination. Moreover, even small and simple information system can depend on many aspects and components (Figure 3), which are necessary for successful project.

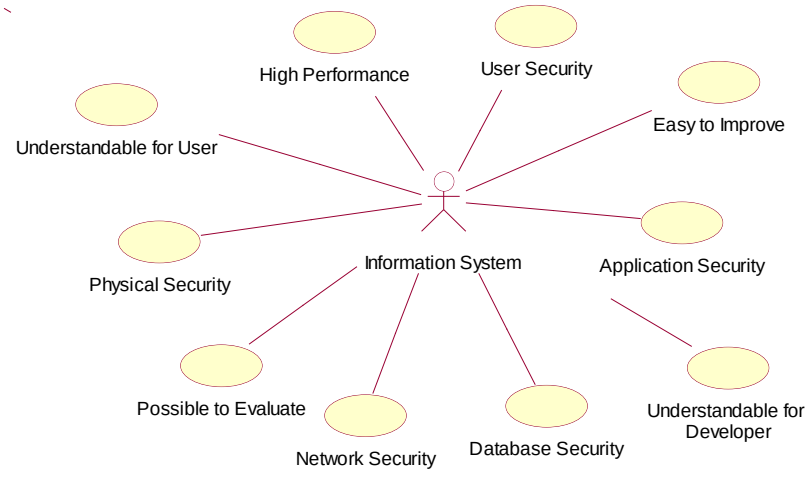

Figure 16. Example Information System Components

Measurement and Analysis Activities

CMMI has the following benefits for projects:

- Helps to collect metrics of data across the organization
- Helps to control project performance
- Helps to establish organizational set of metrics goal
- Helps in transfer of learning

Measurement Objectives

First of all it is necessary to define the goals for measurement. To define, what steps of project are going to be measured. The measurement set should focus on customer objectives, process and project objectives which are critical for performance. Objectives to measure are components of project development. Every step of development will be measured for compliance the standards.

- Documentation
- Requirements
- Risks
- Testing and Security
- Usability

Specifying Measures

If some of the project parts will not meet the requirements of the model, when will be offered measures for elimination the problems.

- Provide a standard, which will help to reduce possibility of mistake
- Calculate the cost of the quality
- Calculate possible schedule variance

Monitoring of Project Risks

Every project, even small have some risks and problems. Even after testing developers cannot be convinced, that system will work without failures, especially considering that risks are not always testable. It can be said that IT risk always exists, whether or not it is detected or recognized by an enterprise.

For this Example information system can be implemented “ISACA RiskIT” framework which helps to identify and manage different types of risks in IT projects. Considering this framework, every risk (e.g., strategic, environmental, market, credit, operational, compliance) is an IT-related risk. In addition, some risks can reduce value or performance of IT project.

From CMMI model, the source of the most risks comes from the system, developers carelessness or from the lack of documentation and uncertain requirements. Some of the typical internal and external risk sources include the following:

- Uncertain requirements
- Not implementable design
- Unavailable technology
- Unrealistic schedule
- Inadequate staffing and skills
- Cost or funding issues

We see the standard can be like a mentor even for experienced developers, helping them to prevent risks on earlier stages of the projects. Another consequence from this part is that CMMI model can be used for project improvement as well, not only quality measurement.

Possible risks considering CMMI and RiskIT requirements are shown in Table 1.

Table 13. RiskIT requirements.

Risk	Risk Example	Affect	Risk Source
Popular trips functionality do not work as it should	Auto input do not work as it should	low	Uncertain requirements
Trip viewing function do not work as it should	User cannot find detailed information for needed trip	high	Bad design
The system does not remember the date of the selected route	User can buy a ticket for the wrong date	high	Inadequate staffing
If there is an error, then system do not notify an error message	The user has no idea about the cause of the error, which can lead to the situation where user does not know how to work with the system	high	Bad design
Absence of help, what could be useful for user	The user has no idea about the cause of the problem , may lead to the situation where the user does not know how to use the services	high	Bad design
Changing the language of the system, causes the loss of page where the user was	Can be the reason of user not loyalty to the system	low	Bad design, inadequate staffing
Departure and destination swapping do not work	Manual input is not convenient for the user	low	Inadequate staffing and skills
Absence of e-mail, while buying a ticket	Loss of ticket	high	Bad design

After pressing a button system should answer in one second	The user can start to get nervous	high	Unavailable technology, inadequate staffing
The content of the system is only in one language	Losing customers, who speak in another languages	middle	Uncertain requirements
Website keywords are not associated with the sale of tickets or the lack of content	Users do not understand the purpose of the system	middle	Bad design
HTML and CSS errors and warnings	Internal changes of the system may bring problems with page appearance	middle	Inadequate staffing
Web page displays differently in different browsers	The differences in design can complicate using of system	middle	Bad design, inadequate staffing
Absence of system mobile version	Impossible to buy a ticket from smartphone	middle	Unavailable technology, Bad planning

Requirements Development and Management

Considering standard "IEEE Std 830" there should be a small documentation before writing a requirements with introduction and overview, but in this article will be analyzed only a part of standard which is connected with different types of requirements.

The basic issues that the requirements writer should address are shown in Table 2.

Table 14. Requirements and their goals.

Requirement	Answers question
Functionality	What the software supposed to do?
External interfaces	How does the software interact with people, hardware of the system, other hardware or software?
Performance	What is the speed, response/recovery time, availability?
Attributes	What is the correctness, portability, security, maintainability considerations?
Design constraints imposed on implementation	Are there any resource limits, policies for databases/software/hardware/language

To evaluate the quality of project requirements CMMI model provides a management process (Figure 4) what helps to manage all requirements generated by the project, technical and nontechnical as well as requirements which are provided by organization.

Requirements which are made using standards and supplemented using CMMI model can bring a lot of benefits to every company. Of course, such approach requires additional work to be done in developing of even small information system. However, with comparatively low investments, the functionality of project can increase considerably.

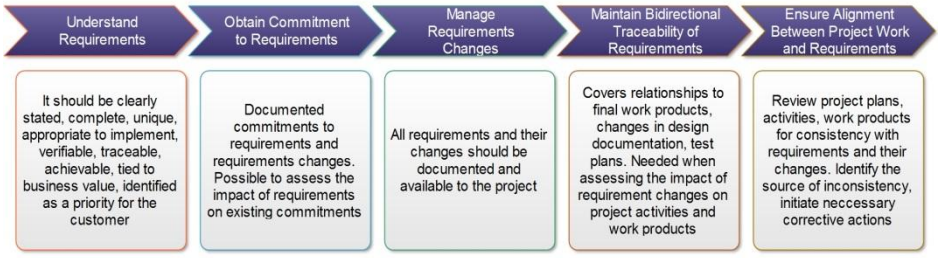

Figure 17. CMMI requirements management process

Testing and Security

The design implementation usually includes unit testing of the product components before sending them to product integration and development of end-user documentation. In September 2013 an internationally agreed set of standards for software testing ISO/IEC/IEEE 29119 was developed. There are currently five parts in this standard, a more detailed description of the standard can be seen below. This set of standards is very detailed and does not insist on using every part of it.

1. Concepts & Definitions - The aim is to create an understanding about all other standards in the series. Introduces the vocabulary on which are based other 29119 standards. Provides definitions, description of the software testing concepts, opportunities to apply the processes.
2. Test Processes - The aim is to define a common process model for software testing that can be used in any software development life cycle. Defines test processes, which can be used for management and implementation software testing in any organization.
3. Test Documentation - There are brought documents that are defined in standard. Organizational: test policy, organizational test strategy. Test Management: test plan, status report, and completion report. Dynamic Test Process Documentation: design-, case-, procedure specification, data and environment requirements and readiness report, actual report, test result, expectation log, incident report
4. Test Techniques - The goal is to create a one standard which will cover test design techniques that can be used in test design and implementation processes.
5. Keyword Driven Testing - This standard helps to describe a test case by using predefined keywords. Such approach will help to easier understand, automate and maintain test cases.

For instance, in our Example information system we have a requirement that first name should not contain nothing else, but small or/and capital letters. This requirement is testable and can be tested using Selenium or other software. Considering ISO standard, after testing it is necessary to create well documented report. Below is shown an example of TEST-INCIDENT REPORT for our information system considering standard ISO/IEC 29119-1: Test Documentation. Test-Incident Report:

- Test-incident-report identifier: TEST-1

- Summary: System allows registration with wrong input
- Incident description:

Table 15. Incident description.

Test_ID	First Name	Last Name	E-mail	Password	Repeat Password
Id_01	Mihhail2	Marenkov	mihail@gmail.com	qwerty	qwerty

- Expected result – error message
- Actual result – page with user data
- Anomalies – result using this input do not answer requirements
- Date and time – 29.01.2014 20:14
- Attempts to Repeat – were used other first names with numbers
- Testers – Mihhail Marenkov

Analyzing this standard, we can make a consequence that using of all standard steps and techniques can be necessary only for big and serious projects, for small information systems there is no need in using all of the standard benefits.

CMMI model helps to verify if all system components were tested as appropriate. Model provides examples of unit testing methods (e.g., statement, branch, predicate, path, boundary, functional testing). In addition, it provides a verification process which helps to ensure that selected work products meet their specified requirements.

Standards Benefits in Real System

In order to make sure that proper use of standardization can help to eliminate problems and improve project quality we choose already analyzed "Tallinn University of Technology" Studies Information System (ÕIS). Usability analyzes and all test results were made independently from our research. The system started to develop in the previous century. At this time, standards were not in wide use in local and other European projects. We provide analysis, how could described issues change if they were made in compliance with standards. Most of issues are connected to usability of the system therefore for analyze we use standard ISO 9241 which provides requirements and recommendations relating to the attributes of the hardware, software and environment that contribute to usability, and the ergonomic principles underlying them.

1. *You want to create a new unique password for TUT systems. Please explore the navigation interface and find this option.*
   From analysis only one person has found a proper menu option from the first attempt. ISO 9241-12 "Presentation of Information" recommends that understandable phrases should be used to improve user's control. For example simply "Change Password" for changing a password. In addition, if tasks have specific steps, when dialogue help should be provided.
2. *You have never got any notification e-mails from the system. Please check your subscription status to university newsletters.*
   Some of the users successfully dealt with this task, however some of them had issues with menu navigation. ISO 9241-14 provides recommendations for the design of menus used in user-computers dialogues. Standard recommends that menu headings should be short. For example "Subscribe for e-mail notifications" can be changed to "Notifications".

3. *You need to get an additional information on the course called "Testing of Digital Systems". Find dedicated web page for this course. The course code is IAF0050.*
   According to user's feedback from analyze, there might be more intuitive and user-friendly terminology used. Standard ISO 9241-110 "Dialogue principles" (2006), gives clear guidance about how should dialogue system react in case if user needs some help. System should prevent giving information to user what is not connected with the successful finishing of the task.
4. *You need to submit the study plan for forthcoming semester. Check if there are relevant instructions provided on how to perform this action.*
   All 5 participants had trouble performing this task. Standard ISO 9241-13 "User guidance" provides recommendations for the design and evaluation of guidance attributes of systems including help, feedback and error management. Standard says that appropriate time for help should be provided and it should not be intrusive. However, now "Help" button is somewhere on top and help for creating a study plan "Abi" is not detectable from the first sight and in fact is not translated into English.
5. *Examination period has been started. Please register yourself to the exam, relevant course code is IAF0050.*
   Considering the performed analysis the average time it took the users to register to an exam is 1:47 minutes which is considerably high result, meaning that the average user should be definitely able to perform this action in shorter time period and with less effort. This problem could not occur if while project designing were used standards ISO 9241-210 "Human-centred design for interactive systems" principle that project can be improved by its evaluation by users.

Other possible usability improvements:

- Larger font size, considering W3C Web Designing standards.
- Button names along with most of the menu options should be renamed according to the actions they perform from user's perspective. ISO 9241-14 "Menu Dialogues" can be used.
- English translations should be added to all the menu items and texts. For this can be used EVS-EN 15038 "Translation services – Service requirements" standard. Standard describes needed professional competences of translators, competences of reviewers and procedures in translation services.
- Additional features for usability improvement from ISO 9241-151 "Guidance on World Wide Web user interfaces". For example, can be added small pop-up window that appears when the mouse pointer is moved over an interaction object and that shows explanatory text or help information. Also, standard advices to use site map with textual or graphical overview of the complete navigation structure of a website.

We think that by solving at least those usability problems (our analysis was not very deep and detailed) ÕIS can become more user-friendly, efficient and will make life of students much better. Of course, standards are not free, but efficiency from using them is obvious. This is obvious, that proper use of standards would ensure avoidance of the problems listed above, because all solutions to ÕIS information system issues were described in detail in standards.

Conclusions and Future Work

In this paper we proposed a study of the quality models and IT standards that covered their definition, implementation, implementation in an abstract and real project, a comparison with other popular quality support tools and an evaluation of the efficiency of standards and quality models lead to the following key conclusions:

1. The standardizing approach and the inclusion of standardization support in the system development offer not only global recognition and opportunity to compete on a larger number of markets around the world; moreover, standards additionally offer opportunity to create a quality and reliable software. Of course, standards do not explain how to create an excellent product, but it provides all needed components and main requirements for creating a good quality product.
2. Since standardization support is a part of systems development and it is available for the entire life cycle of software, the offered standards and CMMI model makes it possible to test, develop and assess the software without stopping the development process.
3. Implementation of standards requires additional work to include needed functionality or documents. Quality assessment using CMMI model do not take a lot of time, if software follows any standards.
4. Standardization saves time by preventing possible failures and errors. This is important in large systems where minor error in the beginning of project may crash the whole system in the end. Continuous using of CMMI model helps to prevent such situation as well.
5. From security perspective standardization and continuous quality evaluation is very necessary, because unsafe system can bring problems to users by sharing their personal data as well as to system developers by local legislation for distribution private data.
6. During analysis we made a consequence that it is strongly recommended that if standards were used in one project stage, when developers should continue using standards in other stages. All stages are very connected; absence of standards in one stage can make them worthless on another.
7. The absence of standards or inability to use them can lead to worthless or duplicated work and can considerably reduce the quality of project.
8. Standards can be a problem for small organizations, because of money. Some of them are pretty expensive. Moreover, there can be need for additional employee for quality management.
9. Different standard certificates (for example ISO certificate) do not guarantee that the company develops projects with a great quality. It just certifies the company. So it can mislead customer that certified company will definitely develop quality project. Therefore, based on research carried out we can make a consequence, that customers should always use quality assessment models to verify the reliability of IT project, despite the certificates of producer.

The standardization support requires attention in every sphere of life. Lack of quality in IT products is the problem that can be found in many European countries.

Further research directions are related to implementation quality assessment model in real project. We plan to expand the scope including different quality models and wider selection of IT standards. Can be made an analysis of different project parts using CMMI, SPICE, Common Criteria or another quality assessment model. Find out if there are any compliances with standards. Specification of CMMI model and basic benefits from standardization in abstract project is the step in this direction. Like we analyzed ÕIS information system in this work, other projects can be evaluated in same way. Moreover, in future researches can be analyzed not only parts of usability, but also requirements, documentation and code structure of real projects.

Quality assurance - aiming to provide safe, reliable, documented and well-structured projects to system users and customers – is unavoidable goal which can be reached using available tools.

Appendix 2.

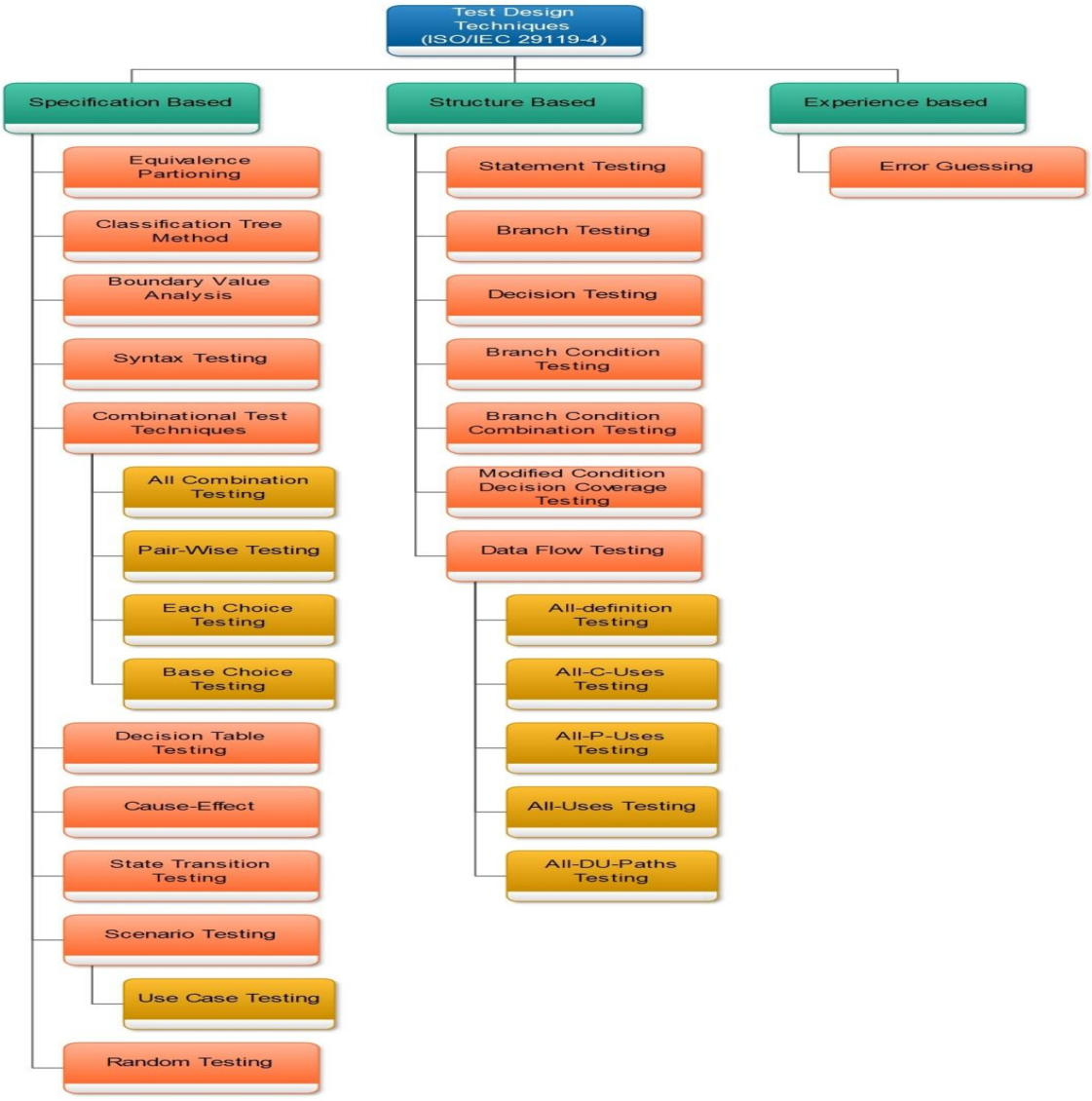

Figure 18. ISO/IEC 29119-4 Test Design Techniques