

UNIVERSITY OF TARTU
Institute of Computer Science
Informatics Curriculum

Tuule Sõber

**Aligning Business Needs with the IT Solution
Design for the Historic Cultural Site
Kreutzwaldi Sajand**

Bachelor's Thesis (9 EAP)

Supervisor: Fredrik Payman Milani, PhD

Tartu 2017

IT-lahenduse ja kliendi vajaduste vastavusse seadmine kultuuripärandi teabekeskonna „Kreutzwaldi sajand“ uuendamise näitel

Lühikokkuvõte:

Käesoleva lõputöö raames kirjeldati veebilehe „Kreutzwaldi sajand“ funktsionaalsust ning andmebaasi mudelit. Valmisid uuenduste funktsionaalsed nõuded, soovitud andmebaasi muutmiseks ning uue süsteemi kirjeldus. Uuriti kuue ärianalüüsi meetodi tulemuslikkust olukorras, kus klient ei ole kindel, mis on uue tarkvaraprojekti tellimise eesmärk.

Võtmesõnad:

Ärianalüüs, süsteemianalüüs, tarkvaranõuded, andmebaasid, Kreutzwald, Eesti Kirjandusmuuseum

CERCS: P170

Aligning Business Needs with the IT Solution Design for the Historic Cultural Site Kreutzwaldi Sajand

Abstract:

This thesis described the existing “Kreutzwald’s Century” website: its functionality and database model. Functional requirements, recommendations for improving and implementing the new, upgraded system were composed. Six business analysis techniques were analysed in the context of this case study.

Keywords:

Business analysis, system analysis, requirements engineering, database modelling, Kreutzwald, Estonian Literary Museum

CERCS: P170

Table of Contents

1.	Introduction	7
2.	Methodology	9
2.1	Business Analysis	9
2.2	Strategy.....	10
2.3	Choice of method	11
2.4	Techniques.....	11
	Benchmarking	12
	Data modelling	12
	Document analysis	13
	Interviews and observation	14
	Process modelling	15
	Root cause analysis	15
2.5	Requirements	18
2.6	Summary.....	19
3.	Case Study.....	20
3.1	Approach	20
3.2	Setting.....	21
3.3	Execution.....	23
	Analysing the current state.....	23
	Problem analysis	34
	Defining the future state.....	36
	Defining the change strategy.....	37
3.4	Summary.....	41
4.	Findings.....	42
4.1	Root cause and business needs analysis	42

4.2	Data processing and verification	43
4.3	Threats to validity	46
4.4	Summary.....	46
5.	Conclusions	47
	References	49
	Appendices	51
I.	Administrator’s functionalities as-is	51
1.1	Data management functionalities	51
1.2	Administrative functionalities	52
1.3	Content management	52
II.	Functionalities of a non-administrative user as-is	53
1.1	General functionality	53
1.2	Timeline.....	53
1.3	Graphical functionality for the timeline	53
1.4	“Kalevipoeg” and Noor-Eesti.....	54
1.5	Collections	55
1.6	Introduction	57
III.	Database model	58
IV.	Functionalities of a non-administrative user to-be	66
1.1	General functionalities.....	66
1.2	Functionalities for a logged in user	66
1.3	Timeline.....	67
1.4	Collections	67
V.	Administrator’s functionalities to-be	69
VI.	List of Problems	70
1.1	Data model.....	70

1.2	Administrators' page	70
1.3	KS website.....	71
VII.	Structure of subpages	73
VIII.	Appearance of KS.....	75
IX.	Proposed changes to the data model	77
X.	Mock-ups of the system to be	79
	Licence	83

Terms and Abbreviations

KS – this abbreviation stands for *Kreutzwaldi sajand: eesti kultuurilooline veeb* (“The Century of Kreutzwald: Website of Estonian Cultural History”) which is the name of the site in question. The website can be found at <http://kreutzwald.kirmus.ee/>.

ECHA – acronym that stands for Estonian Cultural History Archives, one of the four main departments of the Estonian Literary Museum.

BABOK[®] – acronym of “Business Analysis Body of Knowledge”[®], a guide published by the International Institute of Business Analysis[®] that aims to formulate the concepts, techniques and other relevant information of business analysis [1].

1. Introduction

IT projects are an integral part of organisations and businesses today: if done right, they make the company work more efficiently, introduce new possibilities and help create more revenue. Before an organisation can reap its benefits, though, the precarious road of building the system needs to be walked.

According to the Standish Group [2], less than a third of projects in 2015 succeeded: meaning that they stayed within their schedule, budget, and provided the required functionality. The rest of the projects either faced some challenges, such as going over budget, exceeding the allotted timeframe or not having all of the necessary features (52% of projects) or failed to be completed (19%). This means that when an organisation plans to implement a new IT system, it is more than likely that it will cost more than the organisation has planned.

One of the main reasons why projects exceed its budget are the miscalculations and miscommunications between the client and software developers during the planning phase [3] [4]. More often than not, the client organisation does not operate in IT, but still requires IT solutions. Due to this, they might not know what solution would best solve their needs, and even if they do, they lack the necessary knowledge to communicate their ideas.

This means that the planning phase of a project is crucial to its success. In this thesis, a software project's planning phase has been completed. The project is the renewal of the "Kreutzwaldi Sajand" web page¹ which holds data relating to Estonia's cultural heritage. The ECHA who own and manage the site want to have a makeover for it. In order to do that, it is necessary to analyse and document the desired changes to the webpage – this was done by the author of this thesis.

However, it was not clear why the ECHA decided they want to have the current deployed site redone. This was problematic because without having a clear understanding of the business need - why the project is launched -, the project faces a greater chance of exceeding the budget. Furthermore, it would be difficult to decide what kind of changes need to be implemented.

¹ The site can be seen here: <http://krzwlive.kirmus.ee/>.

In light of the previous information and the practical nature of this thesis, the following research question was proposed:

RQ: How can business analysis enable the elicitation of business needs and its alignment to the IT solution? That is, how do the techniques of business analysis help make sure the client gets a solution that meets their needs?

To answer RQ, methods from the field of business analysis were used. Business analysis emphasizes the gratification of stakeholders and defining the problem that must be solved and it can be used whenever there is a need for change within the organisation [1].

Applying the aforementioned methods in this case resulted in the functional requirements for the system-to-be. The current and future states of the database were modelled as well in addition to other recommendations. Finally, analysis of the used methods is presented in light of the research question.

The rest of this thesis is structured as follows. In the chapter 2, an overview of business analysis methods relevant for the research question is presented. In the third chapter, the case study is introduced: the strategy for approaching the case, the background of the case and execution of said strategy are described. In the fourth chapter, the answer to the research question is found. In the last chapter, the work is concluded.

2. Methodology

When need for a change arises within an organisation, it is necessary to analyse the phenomenon to understand its causes and effects. In order to do that, methods from business analysis are used. The field of business analysis is described in chapters 2.1 to 2.2 and the rationale behind choosing it is presented in 2.3. In chapters 2.4 and 2.5, techniques that are used within this thesis are described.

2.1 Business Analysis

The activities conducted during this thesis originate from the field of business analysis. Since the author used methods and knowledge from this area, it is necessary to offer an overview of business analysis and its most important concepts.

The International Institute of Business Analysis (IIBA[®]), the global organisation for business analysts, defines business analysis² as „the practice of enabling change in an enterprise by defining needs and recommending solutions that deliver value to stakeholders“ [1]. In order to enable the aforementioned change, they communicate between the stakeholders: elicit their business needs and limitations, and translate those into the form of requirements that could be used by the software developers to build the product. Depending on the defined requirements and constraints, they propose solutions for the stakeholders along with analyses of the value of each solution [5].

A business analyst needs to take into account the context of the change. More often than not, the change is created through a new or updated IT system, but the work of business analyst can also be applied in other cases [6].

One may ask, why the role of a business analyst is necessary. In order to find the answer, a brief look into the past is required.

According to Whitten and Bentley [6], most system development processes can be divided into the following steps:

1. **System initiation**, during which the problem needs to be identified.

² It ought to be emphasised that even though the field is called *business* analysis, it is not restricted to the private sector. Business analysts are involved with projects in all types of organisations that specify a need for change within the organisation.

2. **System analysis**, which involves analysing the problem and identifying the requirements for the solutions.
3. **System design**: during this step, several solutions need to be proposed and the most suitable one chosen and designed.
4. **System implementation**, which involves implementing the chosen solution and evaluating the results.

Since it is in the organisation's interest to keep using the system for as long as possible, it is important that the maintenance costs be low. However, these costs can be driven high when mistakes are made in the first phases of the development process which then need fixing after the solution has already been implemented. For example, Yourdon [4] has stated that half of the mistakes that are detected after a system's implementation can be traced back to mistakes made when specifying the requirements – before any actual code has been written. What is more, 75% of the cost of fixing these mistakes could also have been avoided if no errors were made during the system initiation, analysis and design phases.

As described by Myers in 1978, “We try to solve the problem by rushing through the design process so that enough time will be left at the end of the project to uncover errors that were made because we rushed through the design process.” [7]

Therefore, making little to no mistakes when engineering requirements is paramount to completing the project within budget. This is what business analysts do – they contribute their knowledge and experience of designing and implementing systems, communicate the needs of the client and capabilities of the developers to each other so that the designed result will be realistic and can successfully implemented.

This thesis deals with the first two phases of implementing a system that were mentioned beforehand: identifying the problem and analysing the problem in addition to eliciting requirements.

2.2 Strategy

A strategy is needed for business analysis: it defines how the client gets from the present situation to the point where their business goals and needs are met by a solution. BABOK[®] proposes the following four activities to perform strategy analysis and find the suitable one:

- 1) **Analysing the current state**: this provides context for the change within the organisation, establishes the existing system and resources that influence the project.

- 2) **Defining the future state:** the result of this phase is a description of what objectives need to be achieved by the project in order to solve the problem or need.
- 3) **Assessing risks:** defining and addressing the potential problems that might obstruct the solution.
- 4) **Defining the change strategy:** analysing the results of the previous three activities that results in potential strategies and choosing the best and most effective one.

These activities³ were performed in this thesis as can be seen in chapter 3.

2.3 Choice of method

Business analysis is a very wide field that is not restricted to implementing IT projects alone – its methods are helpful whenever a change is needed within an organisation (see chapter 2.1).

It became apparent over the first meetings with the client that they did not have a clear idea of what they wanted to achieve with their project nor how to do so – they just knew they wanted to change the KS system. Business analysis was found to be the most suitable choice as it encompasses methods to deal with such a situation where the client only has a vague idea of what they want and need.

Secondly, the scope of the project remained largely unknown as no resources had yet been allocated to the project at the time of writing this thesis. In other possible approaches, the project scope largely determines whether an iterative or a waterfall-type approach could be used. Business analysis, however, does not constrict a project to either, allowing for more flexibility within the initiation and analysis phases of the project.

Due to its flexibility and useful methods, business analysis was chosen as the best method to help with this case study.

2.4 Techniques

In the following subchapters, the techniques used within this thesis will be presented. The author chose to rely on BABOK[®] as the main source for these techniques due to its prominence and the fact that it gives a very comprehensive and *de facto* overview of the industry

³ There was a change to this strategy to tailor it to this case study; see chapter 3.1.

standard. If not specified otherwise, then the information and description regarding these techniques is taken from BABOK®.

Each subchapter will give an overview of a technique and bring out its strengths and weaknesses. In addition, it will be specified how this technique was used within this thesis and why it was chosen⁴.

Benchmarking

Benchmarking means comparing and contrasting the existing practices, products or services of the project at hand against those considered to be leaders in the field. The gaps indicate needs for improvement and also show how to achieve them.

In order to complete benchmark analysis, the following activities need to be performed:

- 1) Identifying the parts of the project that will be benchmarked.
- 2) Identifying the organisations, products or services for comparison.
- 3) Documenting their best practices.
- 4) Analysing the differences between the project and the best practices.
- 5) Suggesting a strategy to implement them.

Even though benchmarking is time-consuming in its nature and does not help to create new, innovative solutions, it can still provide significant improvement for the organisation that uses it.

In this thesis, benchmarking was used to find ways to implement the desired functionalities; mainly to inquire how other websites engage their non-administrative users and display culturally relevant information.

Data modelling

Data modelling describes how data in the given domain is organised: what the different data objects are, how they are grouped, what features they have and what the relationships between them are. The model is most frequently represented visually as a scheme or diagram with explanatory text.

⁴ When choosing the techniques, recommendations by BABOK® were taken into account.

Data is organised into **entities**: groups of objects that have the same features. An entity can be, for example, a hotel, a book, or a warehouse. Entities have **attributes** that hold information about this type of object. An attribute has a data type: it can be numerical, in text form, a boolean etc. For instance, the attributes of a book entity may be the number of pages, the author, and the title of the book. Entities can have **relationships** that specify how some object types are related to others. For example, the book entity and the author entity are related in that an author *writes* a book. One author can have several books and one book can have several authors. Each instance of an entity has a set of identifying features called the **primary key** that is unique to this particular object.



Figure 1. Database model notation.

The notation of the data model can be seen in Figure 1. Entities are notated with their own tables, in which the first section contains the primary key and the rest of the rows are the other attributes. Non-identifying⁵ relationships between entities are marked with dashed lines, one end of which has a strikethrough and the other end branches. This shows the cardinality of the relationship. In Figure 1 it means that one instance of the entity no. 1 can be connected with several instances of entity no. 2, but not the other way around.

In this thesis, the existing database was modelled and improvements to it suggested. Modelling the database in this case study was necessary due to the importance of data and the relationships between them for KS. Showing the user connections between the events, people, and their creations is one of the main goals of KS. In order to do it well, the data model needs to be flexible and up to date.

Document analysis

Document analysis means working through existing documentation about the system or organisation and using the data in those to put together the results of the analysis process. It

⁵ That means relationships where an instance of one entity can be identified without data from another entity. The opposite of this is called identifying relationships but since no identifying relationships are present in the database model of KS, these are not discussed.

can also be used to get context and background for the project or validate the results of it. Even though document analysis may mean relying on outdated or deficient data, it still provides information to the analyst without extra effort.

In this thesis, the following existing documents were analysed to understand the current state of the project and to confirm the information gathered in interviews:

- 1) The IT development plan of ELM 2016
- 2) Project to renew the software of the KS for the public procurement by the Information System Authority (unfinished)
- 3) Report by an archivist of ECHA 2016
- 4) Focus groups interviews conducted by the ECHA about the KS in 2008

Interviews and observation

Interviews involve direct communication between the analyst and the client. Interviews can be **structured** or **unstructured**: the former involves a predefined set of questions whereas the latter means the analyst comes in with no predetermined questions and the discussion evolves naturally. **Semi-structured** interview is a format in which some questions are predetermined, but the interviewer may allow deviation from these according to necessity [8].

Interviews are a flexible tool that enable the analyst to go in-depth to the subject at hand. Since in this case the client did not have a clear need or a problem, extensive interviews were conducted in order to ascertain these and the requirements for the system-to-be. The author of this thesis used semi-structured and unstructured interviews.

Observation means gathering information via observing a process or a stakeholder doing some activity. Observations can be divided into two main types: **active** and **passive** observation. Active observation means that the observer asks questions and clarifications at any point during the process when they deem it necessary. A passive observer does not interrupt the process in any way until it is finished.

During this thesis, active observation was used intermittently with interviews in order to understand the KS system and how it is used both as an administrator and as a regular non-administrative user. It was also necessary to elicit requirements for the system-to-be.

Process modelling

Dumas et al define a business process as “chains of events, activities and decisions” that occur within an organisation during its normal workflow and that bring value to the organisation [9]. Business process models describe these processes: they document and sequence the events, activities and everything else that happens within the workflow of the process.

This thesis used the Business Process Modelling Notation to document the business processes. The main symbols of this notation can be seen on Figure 2.



Figure 2. Notation used for modelling business processes.

The following terms used within business process management have been adopted from Dumas et al [9]. The **start event** is what initiates the process. A **task** is the main unit within a business process. It means one specific and simple action that is taken. Several tasks can make up an **activity**. Data can be displayed in two ways: as a **data object** signifying a certain piece of data or as a **data store** which is a database or any other larger collection of data. An **exclusive gateway** means that from that point on until the end of the gateway is specified, only one of several possible activities can take place. The end of the gateway is notated with the same symbol, but it has several incoming tasks and only one outgoing one. The **end event** signifies the end of the process. **Sequence flow** indicates the sequence in which events take place. An **association** indicates information flow between data and an activity.

The process may be divided into several **lanes** notated by rectangular boxes around the process. These are used to signify different roles and how their tasks fit into the business process.

In this thesis, the process of adding photos into the KS database was modelled to get a better idea of data management with KS and identify possible problems.

Root cause analysis

Root cause analysis means analysing the underlying causes of a problem or effect. For root cause analysis, it is necessary to differentiate between symptoms of the root cause, minor causes, and symptoms. A root cause may be composed of several different problems.

In this thesis, the fishbone diagram, the “Five Whys”, and the interrelationship diagram methods were used.

The fishbone diagram is shown in Figure 3. It depicts the main effect of the problem and organises possible causes and effects into categories. These may include, for example, people, technologies, materials, or practices. Each category goes deeper into the possible causes of the problems within that category.

The fishbone diagram is filled with brainstorming possible categories and causes of the problem. After this, this is analysed in accordance to actual data to identify real causes and solutions.

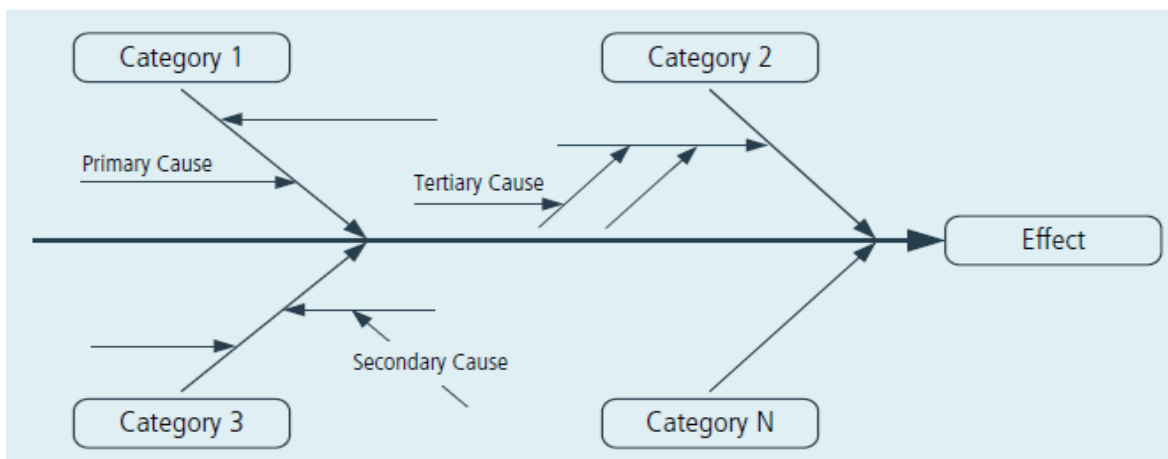


Figure 3. The fishbone diagram [1].

The “Five Whys” is an essential problem identification technique. It means writing down the problem that needs to be solved and then asking “Why do you think this problem occurs?” The answer to this question is captured next to the problem. These steps are repeated until the root cause is identified.

The interrelationship diagram is a graphical representation of all the issues related to the problem at hand and relationships between them: two issues are related when one of them causes the other [10]⁶. It is suitable for analysing complex problems.

Composing an interrelationship diagram consists of the following steps:

1. State the problem about to be analysed.
2. Write down ideas about the problem.

⁶ All information about this technique originates from [10].

3. Organise the ideas so that the ones that might be related to each other are close to one another.
4. Draw an arrow from each idea to another if the former influences and/or causes the latter.
5. Analyse:
 - a. Count the arrows going out of each idea: the more there are, the more likely it is that this idea is the root cause.
 - b. Count the incoming arrows for each idea: ideas with a larger number of incoming arrows are the effects of the problems.

The arrow count (both incoming and outgoing) of each idea can be written next to it on the diagram.

An example of the interrelationship diagram can be seen on Figure 4. It presents the case of an amusement park that has seen an increasing number of unfavourable reviews online. An analyst has mapped all of the problems within the park and composed an interrelationship diagram. When the number of incoming and outgoing arrows are counted, it demonstrates that the bad reviews are an effect of old equipment and understaffing.

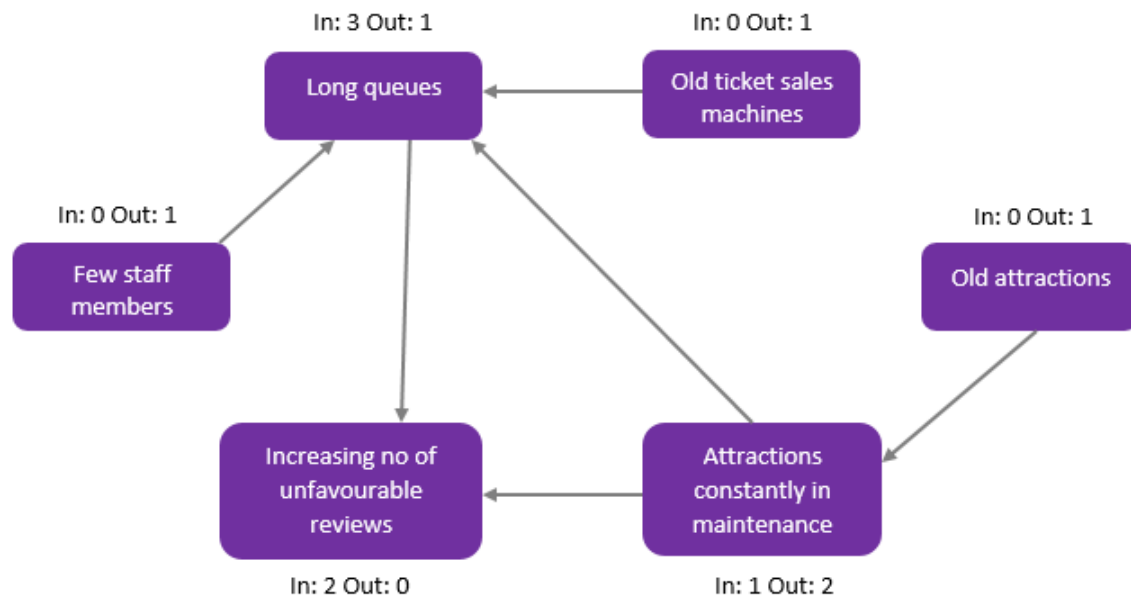


Figure 4. An example of an interrelationship diagram on the basis of an amusement park.

The interrelationship diagram has the advantages of not relying on the biases of the analyst too much: the fishbone diagram puts ideas into inflexible categories and does not allow

depicting causation between the categories. With the interrelationship diagram, this problem is solved.

All of the previous three techniques were used for this thesis in order to map the problems of the KS system, analyse and illustrate them. Identifying the root cause is necessary to make sure that the system that will be implemented will solve the problems its users are currently experiencing and add the functionality that corresponds to the purpose of the system.

2.5 Requirements

In this chapter, requirements are defined and the processes related to handling requirements described.

The ISO/IEC/IEEE 24765:2010 standard [11] that was reaffirmed in 2016 defines a requirement as

- “1. A condition or capability needed by a user to solve a problem or achieve an objective.
2. A condition or capability that must be met or possessed by a system, system component, product, or service to satisfy an agreement, standard, specification, or other formally imposed documents.
3. A documented representation of a condition or capability as in (1) or (2).

[...]

Requirements include the quantified and documented needs, wants, and expectations of the sponsor, customer, and other stakeholders.”

Therefore the requirements for a system must describe everything that the system needs to do with regard to the stakeholders and their needs in a proper manner – they must be quantified and documented. Requirements do not, however, specify *how* the system must achieve these goals.

Requirements for a system are generated in the system analysis phase. They will be used throughout the project: the developers and designers will work in accordance with them and the requirements will be used to verify that the end result has all of the necessary features [3]. The requirements specification and their management are crucial to the project’s success

with several studies identifying these as problematic areas [12] within the software development process.

Gathering, documenting, and managing the requirements specification is called **requirements engineering**. As Aurum and Wahlin [12] put it, “Requirements engineering is concerned with the identification of goals for a proposed system, [and] the operation and conversion of these goals into services and constraints”.

In this thesis, the focus was heavily on **requirements elicitation**: “the process of finding and formulating requirements” [3]. The reader may think, “Wait, this should be easy – why not just ask the client what they want?” In real life, it is not that simple. According to Lauesen [3], one or many of the following barriers present themselves in most of the projects:

- 1) The thoughts of stakeholders do not correspond to the reality. For example, they may think they do not have a problem, and yet the business analyst determines the presence of one. Furthermore, they may not be able to communicate themselves correctly.
- 2) The client is not able to clearly explain their business processes and/or the reasons behind why they perform the activities that they do.
- 3) The client is resistant to change.
- 4) Stakeholders have conflicting ideas about the situation.
- 5) Prioritising requirements is difficult for the client: they may think that everything is essential and ignore the fact that some of those functionalities are “nice to have” or just do not fit into the project scope.
- 6) The problems, requirements, demands and the overall situation change over time.

A requirement is the main unit that is worked with within this thesis and as its outcome, a set of requirements for the new KS system will be presented. This chapter has given the necessary background of requirements, their engineering and management.

2.6 Summary

In this subchapter, the field of business analysis was described along with the specific methods from it that were used. The reason for choosing business analysis as the main approach and the choice of individual techniques was presented. In addition, requirements engineering was described.

3. Case Study

In this chapter, the case study is presented. In subchapter 3.1, the approach - the case study method – is described. In subchapter 3.2, the background of the case study is described and in the final subchapter, execution of the given methods in this case are described.

3.1 Approach

The case study method was chosen to answer RQ. Conducting a case study means studying a contemporary phenomenon in its context [13]. In this thesis, the case study is focused on improvement – the end goal is to compose requirements that will support the further development of a system – and research as it aims to find out how business analysis methods work in a case with the given features.

The case study design, i.e. its steps and their outcomes can be seen on Figure 5. The steps 1, 3, and 4 are the same as described as the business analysis strategy in chapter 2.2. Step 2 was inserted as analysing the problem was crucial to this project because it allowed both the client and the author of this thesis to come to a better understanding of what the KS is, what it does and what it is supposed to do. Risk assessment was omitted because the project lacks a scope: there is no funding or a timeframe which create most limitations and risks for the project.

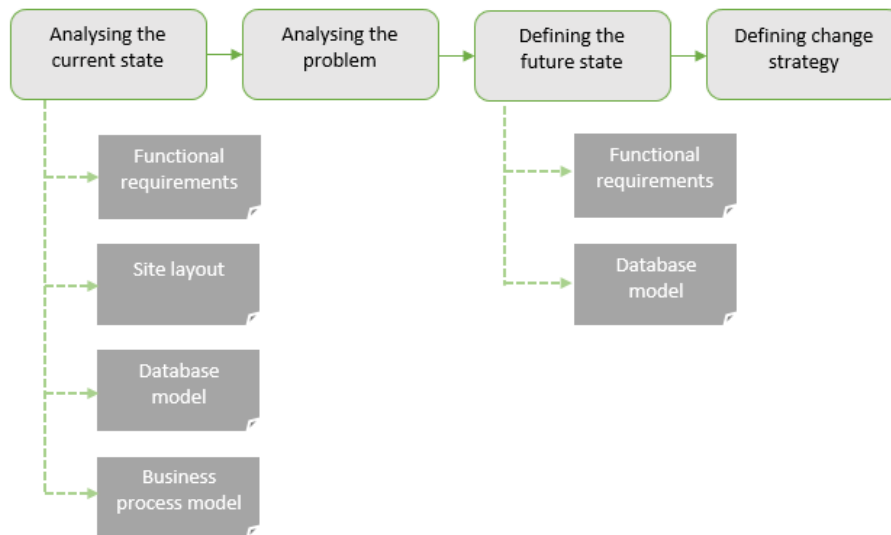


Figure 5. Design of the case study.

The outcomes of the first step were composed using the existing system of KS, documentation analysis and interviews with the client. The outcomes of steps 2 to 4 were composed

using techniques specified in chapter 2.4 and the outcomes of the previous steps. The main form of gathering data was interviews with the client’s representatives. There were altogether seven interviews, each lasting 1.5 to 2 hours. An overview of these is presented in Table 1. The participants from the client’s side were the KS chief system administrator (CSA) who is responsible for inserting data into the KS database and updating the website; and the principal investigator (PI) of KS, who composes and reviews information that is displayed in KS. In one meeting, an archivist (A) from ECHA was also present. In addition to these interviews, one of the employees of the IT company that used to work on KS was interviewed via e-mail.

Table 1. Overview of meetings held with the client.

Meeting no	Attendees	Purpose
1	CSA, PI	Business needs, outcome of the thesis, background
2	CSA, PI	Aim of KS, requirements elicitation
3	CSA, PI, A	Current state (managing photos), requirements elicitation
4	CSA, PI	Requirements elicitation (timelines)
5	CSA, PI	Requirements elicitation (“Kalevipoeg”, Noor-Eesti, Collections)
6	CSA, PI	Requirements elicitation (administrative tasks)
7	CSA, PI	Confirmation of results, future work

In summary, the case study method was chosen to answer RQ. It was designed to comprise of the following steps: analysing the current state, analysing the problem, defining the future state, and the change strategy. There were altogether seven meetings with the client, during which data for all four steps was gathered.

3.2 Setting

The Estonian Literary Museum is a research organisation and archive managed by the Ministry of Education and Research. The Estonian Cultural History Archives is one of the four departments within the Estonian Literary Museum. The aims of ECHA are fivefold [14]:

- 1) Managing and archiving reference books, poetry, and other intellectual works important for Estonia's cultural heritage.
- 2) Conducting research into the national culture.
- 3) Publishing important works regarding the national cultural heritage.
- 4) Participating in the policy creation process within Estonia and Europe through international organisations.
- 5) Cooperating with other relevant organisations both in Estonia and abroad.

In 2003, ECHA organised a travelling exhibition titled “Kreutzwaldi sajand. Kalevipoeg” (“The Century of Kreutzwald. Son of Kalev”) in order to celebrate his 200th birthday. F. R. Kreutzwald was a literary figure whose importance cannot be overstated – he is said to be the founder of Estonian literature and a hero of the national culture [15]. Kreutzwald's most renowned work is composing “Kalevipoeg” - the national epic. He was born in 1803 and died in 1882 – hence the name “Century of Kreutzwald” that references the 19th century.

The exhibition presented Kreutzwald's life and works. Furthermore, it connected Kreutzwald's life to different events of cultural and/or historic importance happening at the same time in Estonia and Europe in general, giving a comprehensive overview of the 19th century and the creation of “Kalevipoeg” in particular within that context.

The exhibition was very popular and its organisers received many requests to put the contents of it up online so that people could access it even after the exhibition itself concluded. The result is the KS system that was created in 2003 by the IT company Laborint⁷ and after that, has been continually supplemented.

Content-wise, the page achieves two main goals:

- 1) The page displays the life events of F.R. Kreutzwald and other major cultural and historical events of contemporary times in a timeline.
- 2) The page enables the user to access historic texts and archive materials such as different editions of the epic “Kalevipoeg”, photos of the era, relevant videos, audio material etc.

⁷ Now Oye Network

The KS webpage is unique in that it shows the user the relations between different objects: people, events, texts etc., both in a timeline format and in the form of displaying the section “Related materials” in every object’s subpage (see page 29).

3.3 Execution

Analysing the current state

Current state analysis was necessary in order to understand and document the existing functionalities, which is needed to propose ideas about how they could be changed and perfected by the system-to-be, especially since the future system would largely be based on the *status quo*. Four aspects of KS were analysed: the KS site layout, functionalities for both the administrative and non-administrative user, the main business process behind using the system, and the data model.

Site Layout

In this subchapter, the current layout of the KS site is shown. Understanding the site layout is important because it demonstrates how information is shown throughout the page. Seeing the layout makes it easier to spot where the logic of the site should be changed and other places that need optimisation. Figure 6 shows the main subpages of the site while Figure 7 and Figure 8 show the structure of the two main subpages of the site: “Kalevipoeg” and “Collections”. The structure of the other subpages can be seen in Appendix VII.

In the figures in this subchapter, the following colour codes are used:

- 1) Green – subpages depicted in green have no further subpages.
- 2) Blue – subpages denoted with blue do have subpages of its own that are depicted in the same or some other figure.
- 3) Red – the page marked with red is the top page.

In Figure 6, the main layout of the KS site is shown. It can be seen that there are five main subpages of the KS site: “Introduction”, “Timelines”, “Kalevipoeg”, “Noor-Eesti”⁸, and “Collections”.

⁸ Noor-Eesti was a literary grouping that was active in the beginning of the 20th century. Their works are accessible in the KS site due to their importance within the Estonian literature scene.

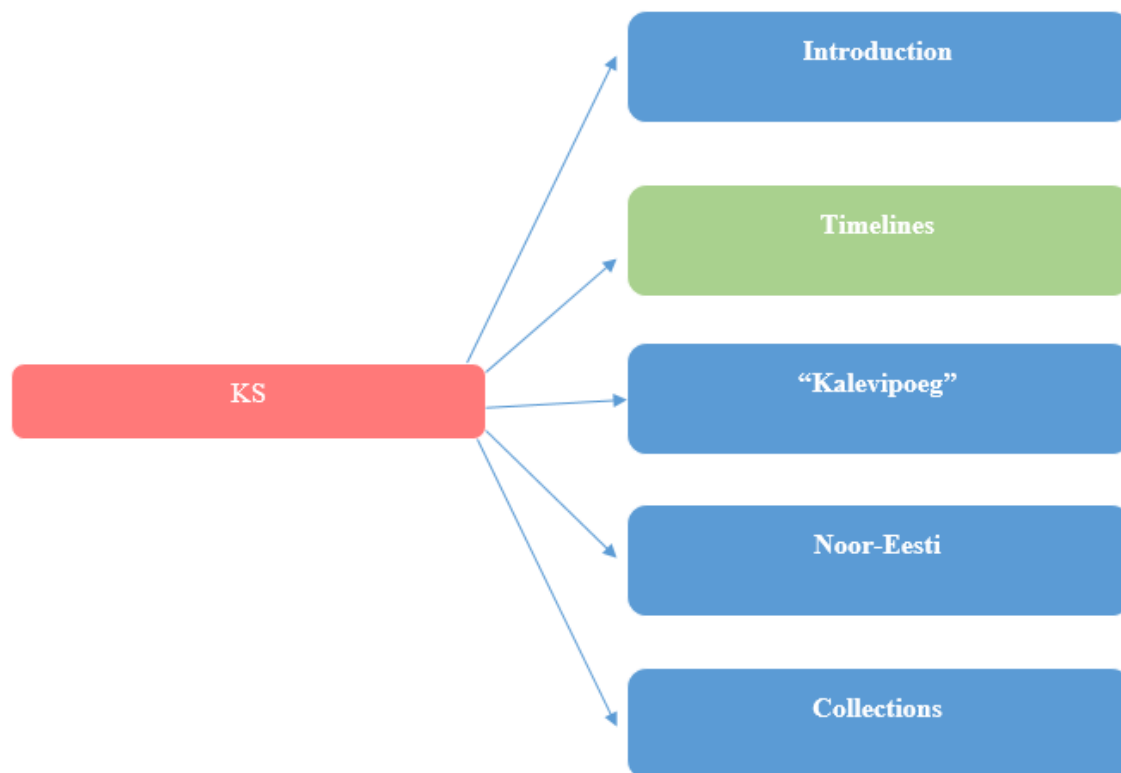


Figure 6. Main layout of the KS site.

The structure of the page “Kalevipoeg” can be seen on Figure 7. The three blue dots denote subpages that have not been explicitly depicted, but that are logically between the subpages mentioned above and below the ellipsis. For example, in this case the ellipsis represent subpages titled “Kalewipoeg, II vihk”, “Kalewipoeg, III vihk” and “Kalewipoeg, IV vihk” because the subpage directly above the ellipsis is the first subpage in the series and the one below it the fifth subpage.

The main aim of the page “Kalevipoeg” is, as is clear from Figure 7, to make different editions of the epic and some related texts available to the reader. Some editions can be downloaded as e-books.

Each book has its own subpage, with two of them being divided into further subpages according to the chapters within that work.

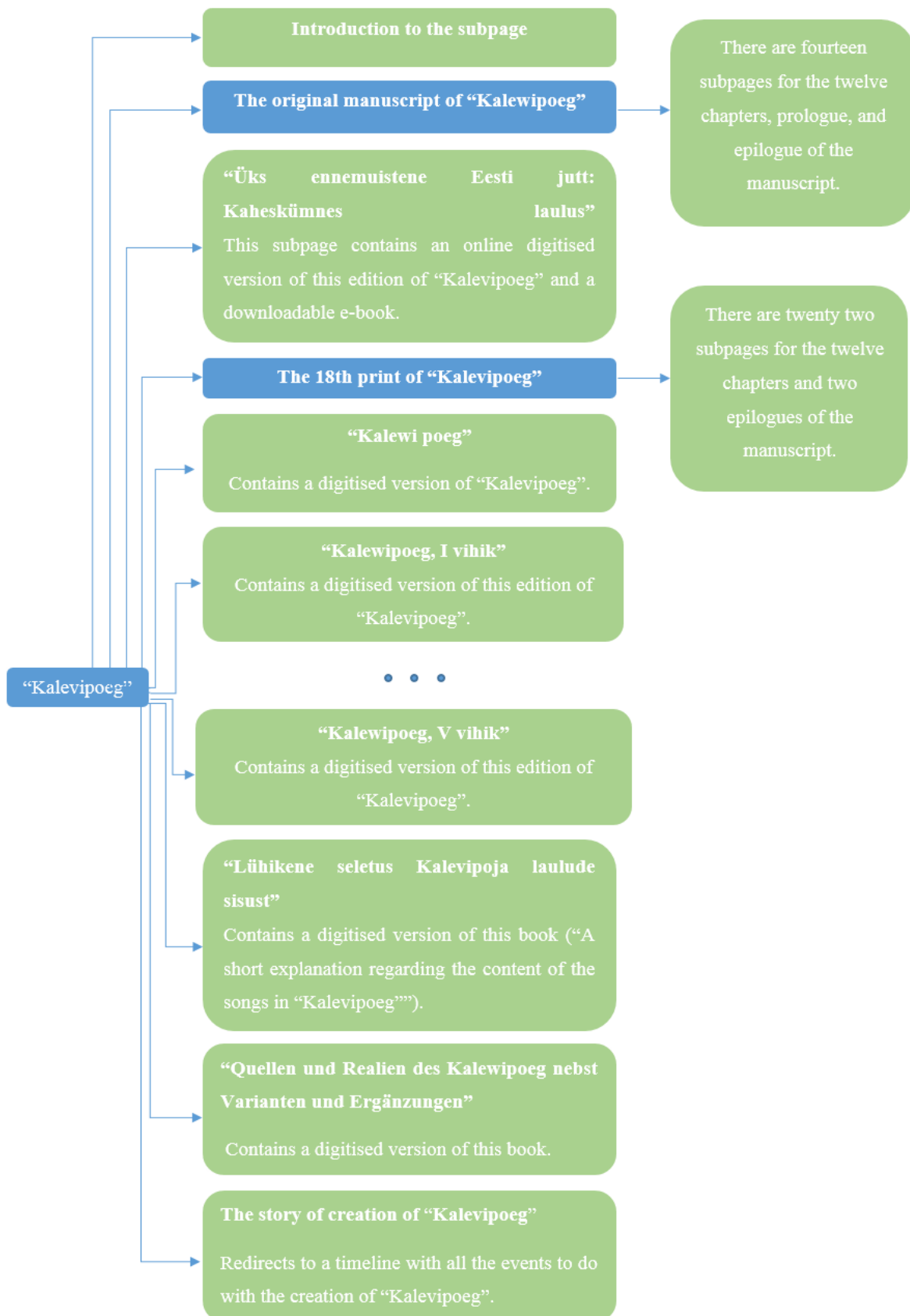


Figure 7. Layout of the page “Kalevipoeg”.

The structure of the subpage “Collections” can be seen on Figure 8. The main goal of this subpage is to present all information that is deemed relevant for this site except timelines. The page is divided into subpages according to the data types represented, e.g. people, audio, video, etc.

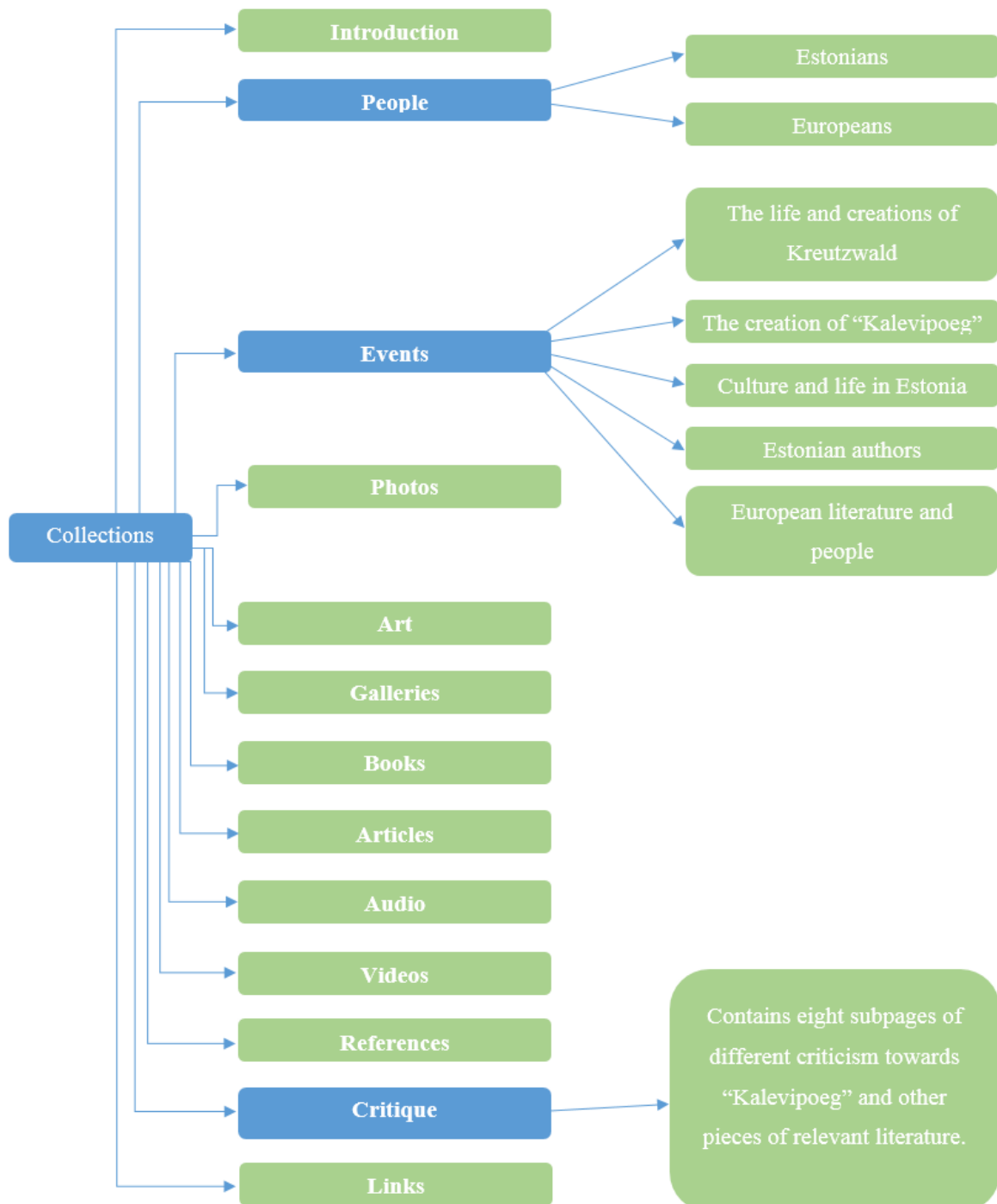


Figure 8. Layout of the Collections subpage.

In addition to the main subpages, there is also the subpage “Timelines” that does not have its own subpages. The page shows an infinite timeline on which relevant events are placed. It shows details about one event at a time and related materials (books, people, etc.), all of which are taken from the Collections subpage. Therefore it can be said that the purpose of the Timelines subpage is to show archive data from the Collections page within its contemporary context.

The appearance of the Timelines subpage can be seen on Figure 9.

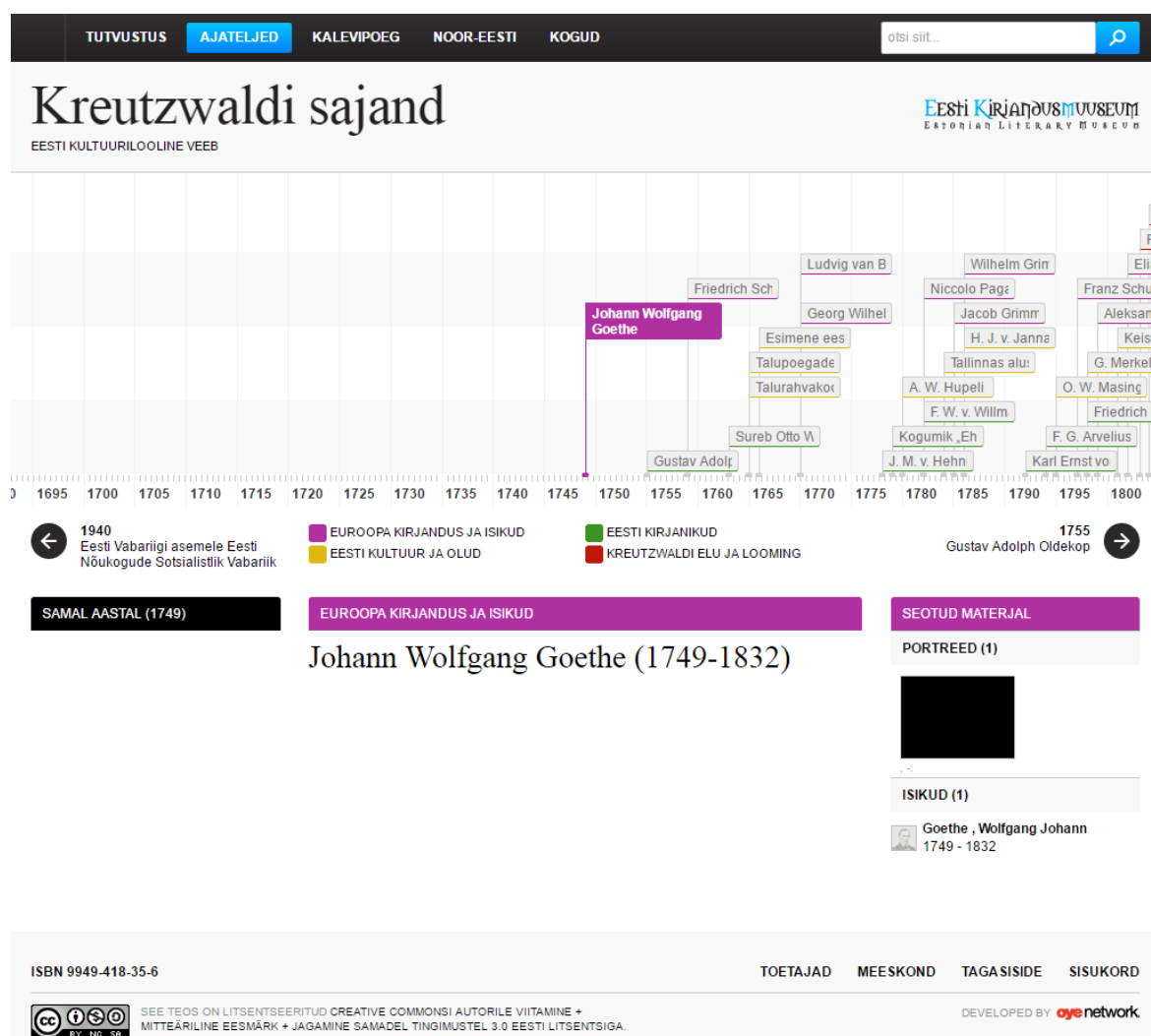


Figure 9. Appearance of the Timelines subpage

As shown in this chapter, the information KS shows is organised into large groups of the literary entity that they are tied to (“Kalevipoeg” and Noor-Eesti). Other information is presented each on their own subpage, organised into categories according to their type (photos, people, events, etc.). The relations between the objects can be seen both through the time-lines and also on each object’s own subpage where other objects related to it are presented.

Functionality

The following chapter describes the functional requirements of the KS site as it is. The requirements are divided into two parts: the functional requirements for the administrator and those for the ordinary user. The requirements for the administrative user can be seen in Appendix I and the requirements for the non-administrative user can be seen in Appendix III. Appearance of some of the interfaces that implement these can be seen in Appendix VIII. These requirements were written based on the existing system and information provided by the people at ECHA.

It can be seen that the tasks that the administrative user must be able to do can be divided into three parts: firstly, they must be able to manage the data in the database; secondly, they must be able to perform administrative tasks on the users of the administrator's system; and thirdly, they must be able to manage the content of the web page. For the first two tasks, there is one administrator's page (see Figure 10), and for content management, there is another (see Appendix VIII Figure 37).

The administrator can create relations between objects – something essential to the structure and purpose of KS – but they are not able to add new object types, that is, they cannot create new database tables or new relations between object types.

On Figure 10, the user interface for the administrator's site can be seen.

The screenshot shows the KS site administrator interface. On the left is a vertical green menu with various object types: Sündmused, Fotode grupp, Fotod, Kunst, Isikute grupp, Isikud, Lood, Teemad, Raamatud, Raamatu tekstid, Raamatu leheküljed, Galeriid, Sündmused ja ajatelljed, Ajatelljed, Ajatelljede grupid, Audio, Video, Tekstid, Vilted, Ajakirjad ja Isikud, and eRaamatud. The main content area is titled 'Sündmused' and includes a sidebar with filters for 'Nimi (eesti)', 'Märksõnad (eesti)', and 'On isik'. Below the filters is a table of events with columns: Nimi (eesti), Kirjeldus (eesti), Alguse kuupäev, Lõpu kuupäev, Märksõnad (eesti), On isik, Viimati muudetud, Raamatud, Ajatelljed, Fotod, and Kunst. The table contains several rows of event data, each with a 'Muuda' button in the sidebar.

	Nimi (eesti)	Kirjeldus (eesti)	Alguse kuupäev	Lõpu kuupäev	Märksõnad (eesti)	On isik	Viimati muudetud	Raamatud	Ajatelljed	Fotod	Kunst
<input type="checkbox"/> Muuda	Eesti Vabariigi asemele Eesti Nõukogude Sotsialistlik Vabariik	HTML	1940-07-21 00:00:00	0000-00-00 00:00:00		<input type="checkbox"/>	2013-09-24 08:28:14	Raamatud	Ajatelljed	Fotod	Kunst
<input type="checkbox"/> Muuda	Andrei Ždanov saabus Tallinna	HTML	1940-06-19 00:00:00	0000-00-00 00:00:00		<input type="checkbox"/>	2013-09-24 09:33:35	Raamatud	Ajatelljed	Fotod	Kunst
<input type="checkbox"/> Muuda	Nõukogude Liit okupeerib Eesti	HTML	1940-06-17 00:00:00	0000-00-00 00:00:00		<input type="checkbox"/>	2013-09-24 09:27:24	Raamatud	Ajatelljed	Fotod	Kunst
<input type="checkbox"/> Muuda	Avatakse tru rahvamaja	HTML	1940-06-09 00:00:00	0000-00-00 00:00:00		<input type="checkbox"/>	2013-10-28 09:55:26	Raamatud	Ajatelljed	Fotod	Kunst
<input type="checkbox"/> Muuda	Juhan Semmle oma kodu	HTML	1940-06-08 00:00:00	0000-00-00 00:00:00		<input type="checkbox"/>	2013-10-28 10:10:09	Raamatud	Ajatelljed	Fotod	Kunst
<input type="checkbox"/> Muuda	IV k/ü „Loodus“ lasteraamatute võistlus	HTML	1940-05-29 00:00:00	0000-00-00 00:00:00		<input type="checkbox"/>	2013-10-30 08:55:43	Raamatud	Ajatelljed	Fotod	Kunst
<input type="checkbox"/> Muuda	„Estonia“ teatri näidendi võistlus	HTML	1940-05-29 00:00:00	0000-00-00 00:00:00		<input type="checkbox"/>	2013-10-30 09:25:01	Raamatud	Ajatelljed	Fotod	Kunst
<input type="checkbox"/> Muuda	Mälestussammas Anton Ounapuule	HTML	1940-05-26 00:00:00	0000-00-00 00:00:00		<input type="checkbox"/>	2013-10-30 10:04:40	Raamatud	Ajatelljed	Fotod	Kunst

Figure 10. Administrator's page for handling event-type objects.

On the vertical menu on the left, all object types are listed – upon clicking one, the administrator is taken to a page where they are able to manage objects of that type. On the right,

there is the management interface for events. At the top of this section of the page, the administrator is able to add new events. The rest of the section displays all objects of this type – their features and relations to other objects.

In Appendix II, the functionalities of the non-administrative user can be seen. As was clear from the layout of the site, the main aim of KS is that these users can see and read culturally relevant information and put it into context: both chronological context (on subpage “Time-lines”) and relational context (embodied by the “Related content” section that is displayed along with each object).

The subpage of one single object can be seen in Figure 11. On the left, the menu for the “Collections” subpage is seen. In the centre of the page, the information about the object is given; in this case, the life and works of the poet and minister Jaan Bergmann. The text contains links to related people, places or other relevant objects. On the right, the section “Related material” is displayed. This section is in turn grouped. The groups in this page are “Places of residence”, “Funeral and grave”, “Family”, “Miscellaneous”, “Ungrouped”, “External links”, “Events” and “Books”. The first five groups contain only photos. This demonstrates clearly the emphasis on putting objects into context in KS instead of displaying them discretely.

The screenshot shows the website 'Kreutzwaldi sajand' (Eesti Kultuurilooline Veeb). The navigation bar includes links: TUTVUSTUS, AJATELJED, KALEVIPOEG, NOOR-EESTI, and KOGUD. The main header displays the title 'Kreutzwaldi sajand' and the logo of the Estonian Literary Museum. The page content is organized into three main sections:

- Left Sidebar (Kogud):** A menu with links to various collections: Juhatus, Isikud (highlighted), Eesti, Euroopa, Sündmused, Fotod, Kunst, Galeriid, Raamatud, Artiklid, Heli, Videod, Vilted, Kriitika, and Lingikogu.
- Central Content Area:**
 - Jaan Bergmann (1856-1916):** A biographical entry. It starts with a summary of his life and work, followed by a detailed paragraph about his education and career. It mentions his time at the University of Tartu and his role as a pastor. A link to 'Jaan Bergmanni teosed e-kataloogis Ester' is provided.
- Right Section (Seotud materjal):** A section titled 'Seotud materjal' (Related material) containing a grid of images:
 - ELUKOHAD (2):** Two photographs of buildings, likely related to Bergmann's life.
 - PEREKOND (1):** A photograph of a family portrait.
 - MATUS, HAUD (11):** A collection of photographs of graves and memorials, with a link to 'VAATA KÕIKI' (View all).

Figure 11. Example of a subpage of one object.

When viewing a book-type object, the different possibilities in which the book can be read are shown as seen in Figure 38 in Appendix VIII. The book depicted is available in three different formats: scanned images, HTML and an e-book in EPUB format. Upon clicking on the latter, a download with the specified file is started.

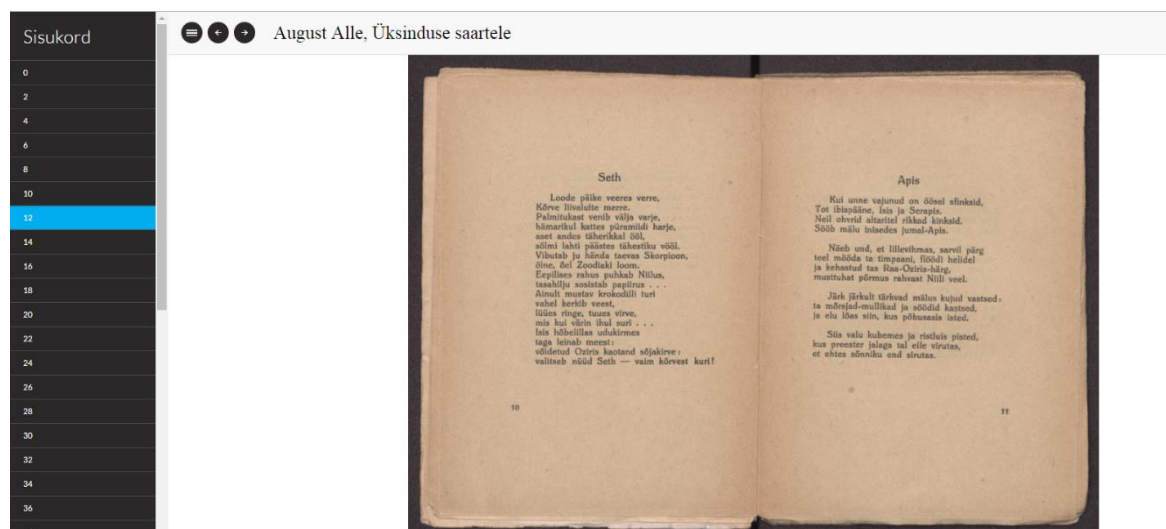


Figure 12. How KS displays a book that has been scanned.

In Figure 12, the user interface for viewing books in the scanned form can be seen. On the left, the table of contents is displayed. It contains the page numbers of the book and upon clicking on one, that page is opened. The uppermost side of the interface contains the buttons for turning pages and one for opening and closing the table of contents (the leftmost button) and closing the interface (the rightmost button). In the centre of the page, the scanned pages are displayed. When viewing a book in HTML format, it is displayed with the same interface (see Figure 36 in Appendix VIII).

There is one more interface: that of displaying photo-type objects. Each photo has its own subpage analogous to the one depicted in Figure 11, the centre of which displays an icon of the photo. Upon clicking on that, the photo is displayed in full-size with information about it and related material (see Figure 34 in Appendix VIII).

There is also a search functionality that conducts a search from the entire database. It can be seen on Figure 35 in Appendix VIII. It is clear from the image that it is currently not working properly, as some search results are displayed without any description.

In this chapter, the functionality of the KS system as-is was described and illustrated.

Data Model

In this chapter, the current data model for KS is described as it was one of the client's desired outputs. In Appendix III, the database model can be seen. The database was modelled as a relational database using existing XML and SQL scripts. There is 1.45 terabytes worth of data in the database.

There are 44 tables altogether. One of them holds information about the users of the administrator's site (*erm_users*). The other 43 are connected to the information displayed in the KS and can be divided into two broad categories: tables that represent a specific entity and "connection tables" that connect two tables in order to translate all M:M relations into 1:M ones. The former category contains tables such as *books*, *persons*, *video*, *audio*, *events* etc. The latter contains tables like *eventsaudio*, *personsvideo*, *eventsbooks* etc.

The data model is solid and works well with this system. Some ideas for improvement are offered in chapter "Database model improvements".

Business Processes

The business process for inserting new photos into the KS database can be seen on Figure 13. The process of inserting photos was chosen to model due to its complexity compared to the processes of inserting other types of data into the database.

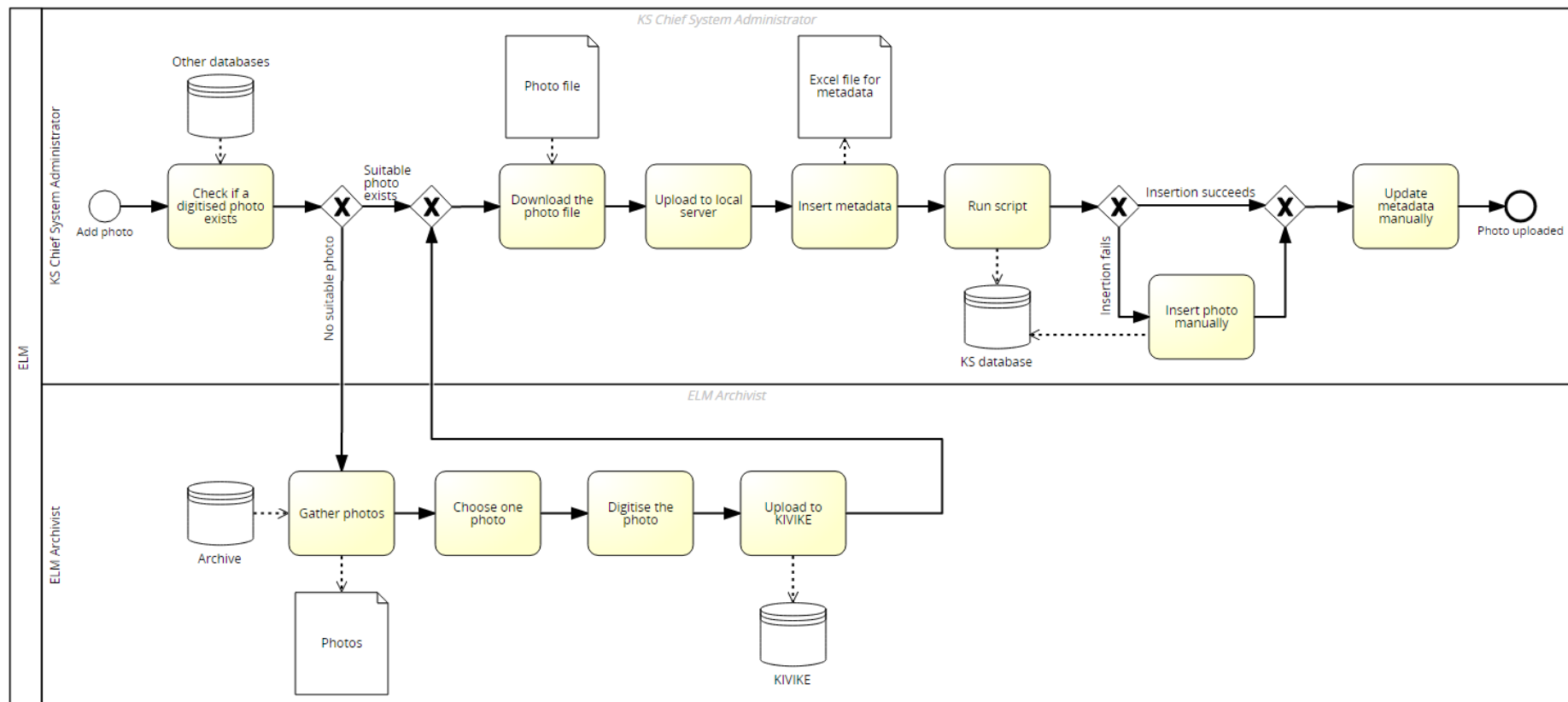


Figure 13. Adding photos into the KS database.

There are two people who are relevant to this process. The first one is the chief system administrator of KS and the second is an archivist from the ECHA. When the KS staff decide to add a new photo to the site, they first check the following existing databases to see if they already have a sufficiently similar photograph:

- 1) The Estonian Literary Museum's database KIVIKE
- 2) The Estonian National Archive's database of photos FOTIS
- 3) The Estonian Museums' Information System MUIS
- 4) The Estonian National Archive's database SAAGA

If a suitable photograph is not found, then the archivist is included into the process. They will find all the relevant photographs from the ECHA's archives. If there are several copies of the same photograph, they will retrieve all of them and choose the one that has the best quality. Then the image will be digitised. The archivist will check the quality of the digitised file and then upload that along with the metadata of the photo into the ECHA's database KIVIKE. Once this has been done, the KS system administrator will retrieve the digitised photo in JPG-format from KIVIKE and download it.

If a suitable photograph was found from KIVIKE or some other database, then the administrator will also download the image in JPG-format.

The KS system administrator will then insert metadata for the photo into a XLS-file and put the photo into the local server. The metadata includes the name of the person who this photo relates to (whether they are in the picture or the picture depicts something relevant to that person). They will then run a script that uploads the photos that need to be added along with their metadata.

The script will not upload the photos when the metadata is not correct: namely, the script will try to create a relation between the photo (database table *scans*) and the person whose name has been put into the metadata (database table *persons*). If a person with this name does not exist in the database, the photo will not be inserted into the database. In this case, the administrator will have to insert this photo manually along with the person.

The script will only create the relation between the photo and a person. All other relations (e.g. with events, galleries, etc.) need to be inserted manually by the administrator.

After this has been done, the picture exists in the database and is also displayed by the KS site to all users.

Problem analysis

In this chapter, the analysis of the problem and the business need of the ECHA is presented: why they wanted a new system. This analysis was conducted based on the outcome of the first step and data gathered through interviewing the client. The methods for performing this analysis were described in chapter “Root cause analysis”.

The problem that was first presented to the author of this thesis was that the code base for the system was old (created in 2003, had several functionalities added to it and the design changed since then; the last change to the system was made in 2012). The client also claimed that they had been told by the IT-company that the functionality that they wanted to add to the system could not be done due to outdated code and the site architecture.

In the first interview, the business need for interactivity was identified in addition to that. Namely, the client said they wished KS would be more interactive and allow people to create something themselves instead of being just a passive consumer of information.

The IT-company who used to work on the site was contacted after the initial meeting with the client. They said that the functionality could, in fact, be added. The fact that the code is old means that

- 1) The user experience, especially of the administrator’s site, could be improved significantly.
- 2) In the light of the new functionality that the ECHA wishes to add, it would be better to update some or all of the technologies and platforms used in KS as they would make the system more efficient to program, manage and use.

After this, it was taken to the client again: what is the purpose of the new system? During the interview it became apparent that the main reason is related to the purpose of the ECHA. Namely, the ultimate goal is for the cultural heritage of Estonia to be preserved. In order to do that, young people need to be interested in and engaged with the material. To achieve that, it needs to be presented in an interactive manner.

To compactly and comprehensively present the given information and data gathered from the interviews, the fish bone diagram analysis was performed.

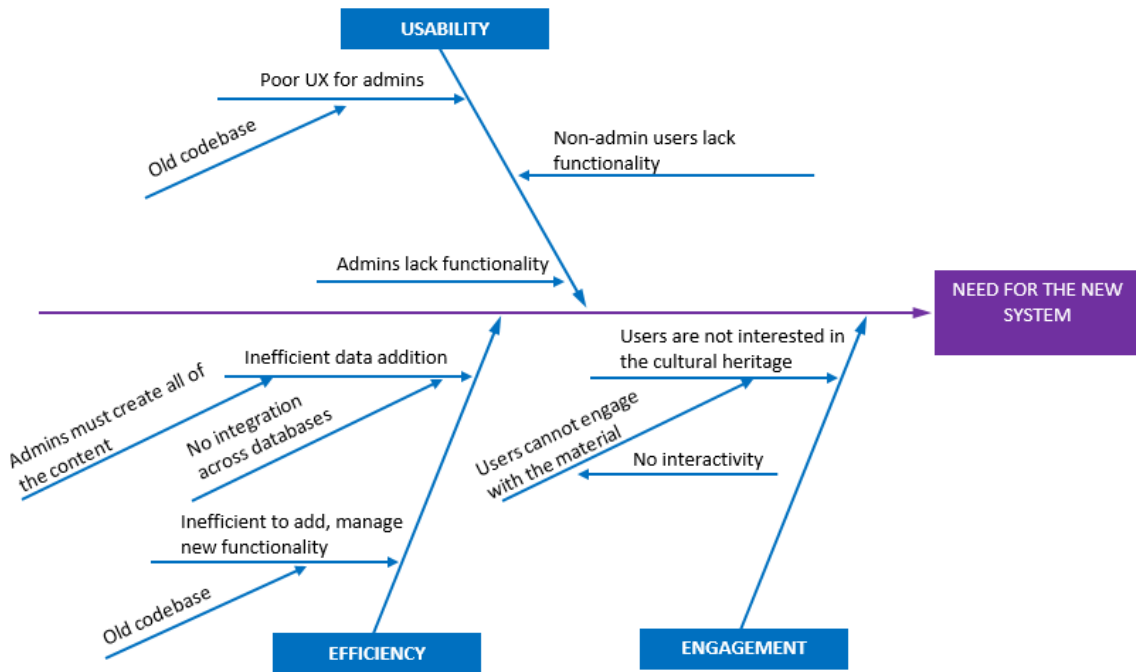


Figure 14. Fishbone diagram of the problems regarding KS.

The main causes for requesting a new system are problems with the current system's usability, efficiency and engagement (see Figure 14). The most important one is the latter: that is the root cause to the KS existing in the first place, therefore this problem is what needs to be most addressed by the new system.

An interrelationship diagram was also composed and it can be seen on Figure 15. The issues were adapted from the fishbone diagram. Analysing the diagram reveals that there are three root causes:

1. *Old codebase* and *missing functionality* cause all of administrators' problems.
2. *Missing functionality in terms of user engagement* is a root cause for all users, increasing the workload of administrators and meaning less interest and engagement for non-administrative users.

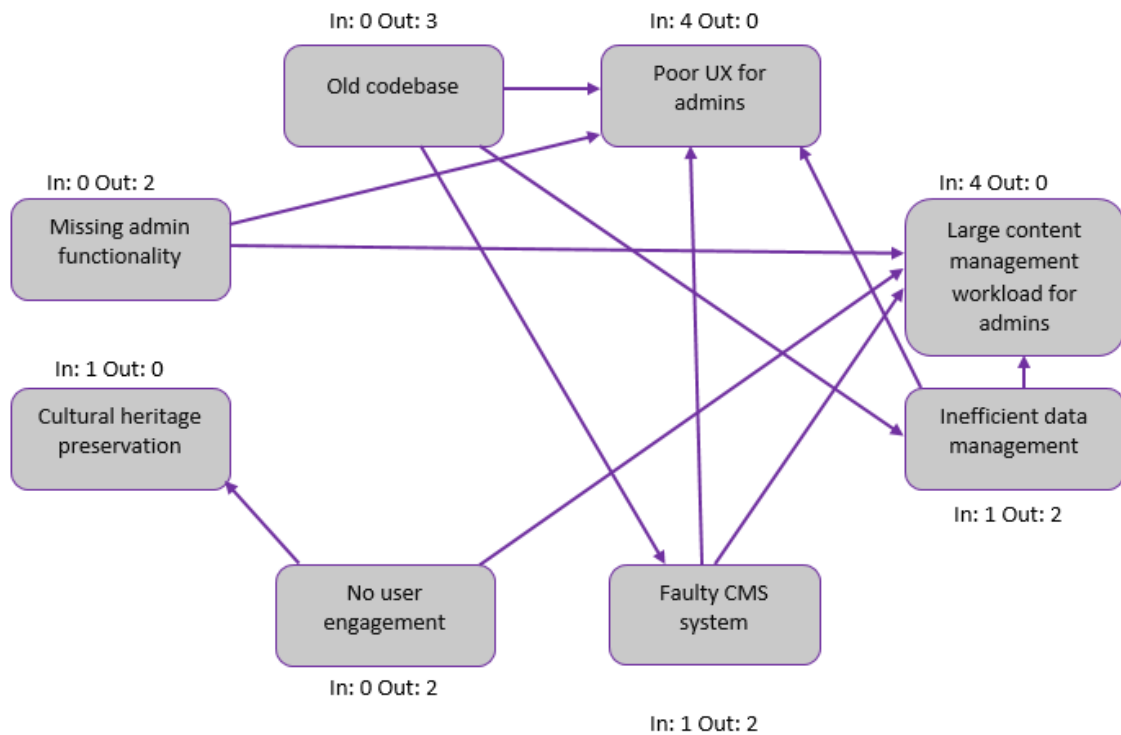


Figure 15. The interrelationship diagram about the issues of KS.

In addition to identifying the root problem, throughout the interviews and observation, more minor problems came to light. Most of them concern faulty implementation or bugs and not the functionality of the site, while a few highlight problems with the architecture of KS. Since documenting these will be important as the new system will be based heavily on the old one, they are presented in Appendix VI.

The fundamental business need of ECHA is to make cultural heritage accessible and interesting. Old codebase and missing functionality cause most of the problems within KS as it is today. During the course of problem analysis, a non-exhaustive list of bugs and flawed implementation was composed as well.

Defining the future state

The future state must describe what is to be achieved. It answers the question “what must the system allow the client to achieve?” Note that the change strategy will deal with the question “How will the system implement the future state?” In order to define the future state, the outputs of the previous two phases were considered.

The future state must be subject to three conditions. Firstly, the new system must preserve the current functionality; this is necessary because the current functions offer data regarding

the Estonian cultural heritage, the preservation of which is one of the aims of ECHA. Secondly, it must decrease the workload of the ECHA employees when managing KS and inserting, updating or deleting data within the system. As it turned out in problem analysis, missing functionality and old codebase create a lot of problems and need for manual labour (for tasks that could be automated) when performing administrative tasks. Thirdly, the new system must help preserve Estonian literary heritage as this is what ECHA aims to achieve. Even though this concept is abstract in nature, in this case there is a specific meaning as per client's vision and experience. Namely, preservation will be achieved through engaging users with the content: it must allow them to "play around with it" in some way that is active (active activities of websites include writing, editing, deleting, creating something new) rather than passive (reading, scrolling).

Metrics

Some metrics are offered by the author of this thesis as a way to vet that the new system follows the aforementioned conditions.

1. The system must perform all functions defined by the functional requirements of the system as-is⁹.
2. The time it takes for an administrator with reasonable experience with KS to insert data into the system must be 25% shorter.
3. Users stay on the page on average 20% longer as they do with the current system.
4. Within a year of the launch of the new system, at least one gigabyte of the data on KS must be user-generated or -submitted.

Defining the change strategy

The future state of the KS is a system the same system that has new functionalities. The techniques of benchmarking, interviews and observation as they are described in chapter 2.4. The future state was defined through the following outputs:

- 1) Functionality for the future state.
- 2) Benchmarking to suggest ways to implement the future state.
- 3) Suggesting improvements to the database model.
- 4) Mock-ups.

⁹ They do not have to be implemented in the same way as they are in the current system.

These four will be discussed in the following chapters.

Functionality

Using the aforementioned techniques, the functional requirements for the system-to-be can be seen in Appendices IV (for the non-administrative user) and V (for the administrative user). These contain additions to the functional requirements of the system as-is due to the fact that the system-to-be needs to have new functionality while preserving all that it does so far. The requirements were composed according to the client's vision of the system-to-be.

For the administrative user, the main differences are the ability to change the database themselves and exporting data from other databases into the KS. For the non-administrative user, the main difference is adding a user management system with the ability to create a profile, log into and out of it, and some functionality that is only available to logged-in users. This includes the possibility to create and publish objects of one's own making, such as adding photographs, events, etc. Furthermore, creating relations between objects, including devising one's own timelines is also possible.

These functionalities mean that for the administrator, the life gets easier as a lot of manual work and duplication is removed. The new functionality for the non-administrative user means helping solve the business needs of the ECHA: interactivity and the ability to be an active consumer of information rather than a passive one means more engagement with the material; playing around with it makes it more interesting.

Benchmarking

In this subchapter, ideas on how the overall aims of the ECHA can be achieved via the new system are presented. More specifically, the technique of benchmarking is used to visualise how other digital libraries and museums present data.

The first part of the website to be benchmarked is the timeline. The timeline on the World Digital Library¹⁰ homepage that can be seen on Figure 16 was created using the TimelineJS tool. It is in general similar to the KS one:

- Events are divided into lanes according to their type.

¹⁰ <https://www.wdl.org/en/sets/world-history/timeline>

- Only the beginning of the description is displayed on the timeline.
- The design of the timeline is similar.
- The timeline has a zoom functionality.
- The details of an event are displayed next to the timeline.
- The user can navigate between events one by one.

There are also some elements that the KS timeline is missing but would benefit from:

- The duration of an event is displayed on the timeline itself as a blue line.
- Each event has its own URL: this makes it easier to share events.
- Fading out events that are not currently active helps grasp the timeline better.

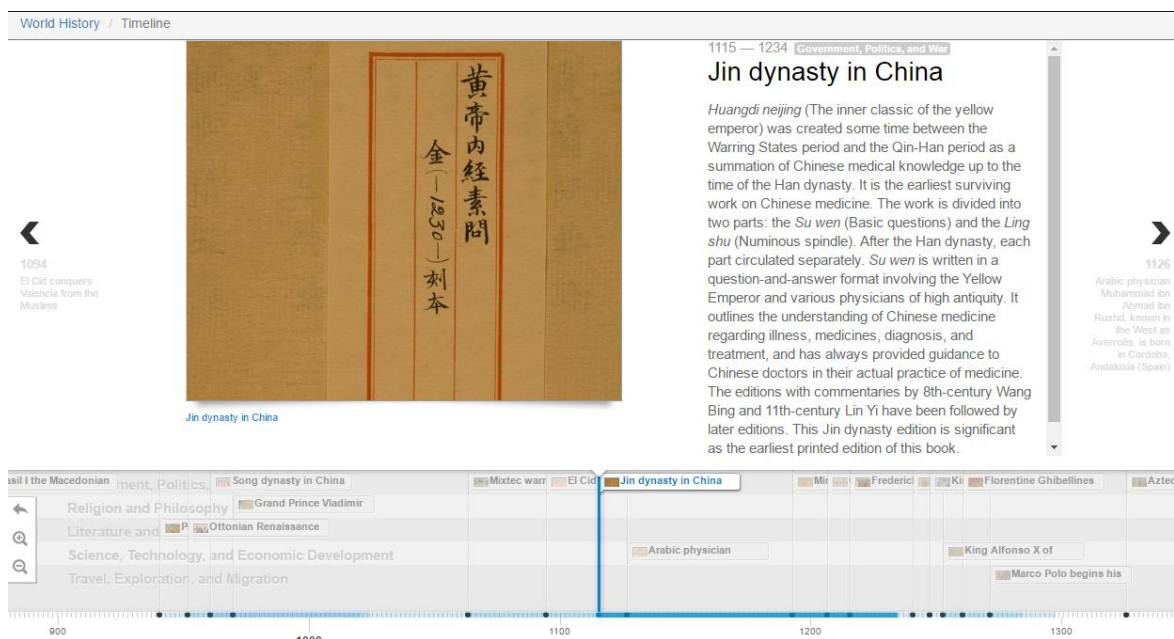


Figure 16. The timeline on the World Digital Library homepage.

The Metropolitan Museum of Art has a website for children: MetKids¹¹. Among other things it contains a “Time Machine” feature which is essentially a search box: the user can choose a time period, a geographical region and an abstract idea; after this, they are shown pieces of art that fit these criteria. Upon clicking on one of the search results, the following are displayed:

1. Information about the piece
2. An “Imagine” section which contains a question related to the piece

¹¹ <http://www.metmuseum.org/art/online-features/metkids>

3. A “Create” section which suggests fun and creative activities for kids related to the piece

There is also a MetKids blog, where children’s handiwork is often displayed in addition to posts targeted to all youth.

Even though coming up with questions and activities is time-consuming, this would be one way to help ECHA achieve its goal of both allowing users to create content and make KS interactive.

Since the idea of allowing users to create their own content in a museum’s website is not widely used [16], not much benchmarking material was found. Most websites that aim to display collections of museums simply present the data. This means that the new version of KS would be truly innovative.

Database model improvements

In this chapter, improvements for the database are suggested based on the outcomes of database modelling.

The main suggested improvement for the database is the ability for the administrators to add new tables, that is, object types; define, update and delete relations between tables, and edit and delete existing tables. The alternative to this is just altering the database model every time that necessity arises, but since the ECHA lacks an in-house IT department, conducting a public procurement every time a change is needed will be a waste of resources.

The proposed changes are adding three tables: *manuscripts*, *letters*, and *personsbooks*. The rest of the changes facilitate creating these three tables. The tables, along with their attributes and relations can be seen in Figure 39 in Appendix IX. Each table row represents one database table to be added.

The following “cosmetic” changes are proposed:

1. Delete attribute *personid* from table *books*.
2. Rename table *booktextpges* to *textpages*.
3. Rename table *bookpages* to *pages*.
4. Rename attribute *book_id* from table *bookpages* to *source_id*.
5. Rename attribute *booksurls* in table *booksurls* to *booksurls_id*.

These changes facilitate needs the presence of which became apparent during meetings with the client. They would allow new types of data to be added to KS (adding tables *letters* and *manuscripts*) and alter the database so that it would correspond better to reality (allowing a book have multiple authors).

Mock-ups

Two mock-ups were created to support the image of the future system: one for creating relations between objects for an administrator (consists of five images) and one for creating their own timeline for non-administrative logged in users (one image). The mock-ups can be seen in Appendix X.

3.4 Summary

In this chapter, the structure of the case study was presented: the case study approach was described, the setting of the case study (upgrading the KS system) was briefly introduced, and the execution of the study described. The latter was described in four aspects: it was described in its present state, the problems with the present statue were presented, along with what the future system should do and how this might be implemented.

4. Findings

In this chapter, the findings of this thesis are presented in regards to the research question. There were two main findings: firstly, identifying the underlying causes and business needs is essential and business analysis techniques are most helpful in this regard; secondly, for requirements elicitation interviews are invaluable when gathering data and all techniques with visual outcomes are helpful in verifying the results with the client. The first result is discussed in detail in subchapter 4.1 and the second one in 4.2.

4.1 Root cause and business needs analysis

As seen in chapter “Problem analysis”, the analyst was presented with several problems and needs: at first old code was thought to be the problem, later lack of interactivity and the need to preserve Estonian cultural heritage were pointed towards. It must be noted that all of these were problems and needs – but causal relationships between them needed to be identified. Furthermore, different problems would be of different priority to the client. The technique that was used to gather all of this information was interviewing: it was also during an interview when one of the main business needs – preservation of heritage – was identified.

After collecting this data, it needed to be analysed to identify the real causes to problems the administrators were experiencing. To do this, root cause analysis techniques were used, namely the fishbone diagram and the interrelationship diagram. The most valuable results were created by the interrelationship diagram. Specifically, old code was found to be a major problem even though it was previously thought that having updated code would be just “nice to have”. In addition, these results confirmed that more functionality for all users would help achieve ECHA’s goals and decrease the workload of the administrators.

The time it took to identify the real need for a new IT solution means that this case study is yet another example of a client who wishes to develop a new IT solution, but is not certain why. It is important to identify the cause in order to plan the future state, change strategy, and assess the requirements without going over budget (see chapter 1).

It should be noted that eliciting requirements was already underway when the final root causes were identified (see Table 1 in chapter 3.1). This illustrates the benefits of iterative approaches during analysis, as there is time between meetings to process the data gathered, and elaborative questions can be asked during follow-up interviews.

4.2 Data processing and verification

Interviews were the most useful tool for eliciting requirements. Document analysis helped prepare for interviews, confirm what was discussed with the client as facts, and occasionally yielded additional data, but could not be relied upon on its own.

When discussing the results with the client, all techniques that resulted in something visual (data modelling, fishbone and interrelationship diagrams, mock-ups) were the most useful ones – it illustrated points that otherwise would be too obscure or difficult to grasp and gave them something concrete to base their decisions on and to give feedback to. Furthermore, this helped reduce misunderstandings between the client and the analyst as instead of abstract concepts, a concrete idea was presented in terms that a client with little IT-knowledge can comprehend.

It should be noted that even though benchmarking meets these conditions, it was not found to be useful in this case as there were few other museum or library websites that allow the user to do anything besides just viewing the collections.

The usefulness of some techniques cannot yet be fully assessed. The modelled business process helped verify the time-consuming nature of managing KS and data modelling provided documentation for the database, but when it comes to eliciting the business needs, describing the future state and the change strategy, the results of the aforementioned two techniques were not used. They will, however, most likely be turned to when implementing the solution begins in order to make sure the new system is more time-efficient and preserves the data model.

The used techniques, their benefits, and negative features identified in this case study can be seen in Table 2.

Table 2. Assessment of business analysis techniques in the given case study.

Tech- nique	Was used for	Pros	Cons	How it helped identify the need
Bench- marking	Aligning the understand-ings of the analyst and the client in defining the change strategy	Clear understand- ing of what is dis- cussed Gave new ideas	Very few websites do something sim- ilar to what the cli- ent desired	-
Data modelling	Creating the data model of the KS data- base based on existing XML and SQL scripts in cur- rent state analysis	Gave a clear over- view of how the data is organised; improvements could be suggested based on it	Time-consuming (44 tables) Unclear scripts	Helped identify problems with the database, but not with the main business needs
Document analysis	Intermittently to understand the existing system and the wishes of the client	Gave background information that would otherwise be impossible to get, including past ideas for improvement	Time-consuming Sometimes did not yield anything useful for time spent	Confirmed the main business need for cultural heritage preser- vation

Inter-views and observation	Main form of data gathering throughout	Could ask for additional information Flexible regarding discussion points	Time-consuming Interviewees did not always know the answers to questions Hard to always find a suitable time for all participants	Identified the main candidates for business needs
Process modelling	Analysing the current state	Detailed; helped identifying problems Quick Clear to grasp	There were too many processes to model	Identified a problem (time-consuming data management) that was not a root cause
Fishbone diagram	Identifying the root cause	Illustrated all of the issues Identified connections between problems, making it easier to distinguish causes and effects	Some issues fit into several categories Result depended on what the analyst <i>thinks</i> causes the problems	Helped distinguish causes from effects Helped map issues with the system
Interrelationship diagram	Identifying the root cause	Identified causes that would otherwise be overlooked	Somewhat dependent on the biases of the analyst Image itself became cluttered	Resulted in a list of root causes not influenced by the analyst's biases

All in all, the answer to RQ – how can business analysis help align the business needs and the IT solution in a case where the client is not entirely sure about what they want – is twofold. Firstly, identifying the root causes and needs is essential. Spending as much time on it as needed helps make sure that in the end, the client does not get a system that does not solve their problems. Secondly, the specific techniques that are recommended are interviewing the client and processing this data with the interrelationship diagram. These should be supplemented by techniques that result in something visual to help make sure the client knows what they get and can offer feedback.

4.3 Threats to validity

The following threats to the validity of the results of this thesis were identified:

1. Problems might have been overlooked.
2. There might be techniques more suitable for these types of case studies which were left out of this thesis.
3. The results depend on the analyst's subjectivity as there are no definitive answers.

In order to decrease the effect of these threats, the following actions were taken:

1. There were several iterations over discussing the same system components and their problems.
2. All techniques described in BABOK® – the comprehensive field guide for business analysts – were carefully examined to make sure the most suitable ones were chosen.
3. The results were run by two representatives of ECHA to ensure their validity.

Due to preventative measures taken, the results of this thesis is believed to be valid.

4.4 Summary

In this chapter, the answer to RQ was presented: in a case where the client is not certain about their needs and problems, thorough problem analysis is needed. For this, interviews and root cause analysis techniques are recommended. In order to align the results of this analysis with the ideas of the client, visually representing information is to be preferred. In addition, each technique used was analysed in terms of their positive and negative features.

5. Conclusions

The goal of the KS website is to share information, art, books, and other knowledge about the Estonian literature landscape in the 19th century. Not only is its objective to enable researchers, teachers, and students access information, but it also aims to engage its users by putting people and events into their contemporary contexts.

When this thesis topic was first approached, the outcome seemed clear: requirements for an updated KS system ordered by ECHA. After taking on this task, it became apparent that the client only had a vague idea of what the system should do and why it should be implemented in the first place. Building an IT solution without identifying this will mean more costs for the organisation in the future because it is not verified whether the solution will actually solve the problems at hand. This brought the field of business analysis into play and the research question “How can business analysis help align the business needs and problems with the IT solution?” was proposed.

In order to answer the question, techniques from the field of business analysis were chosen as per their suitability and usefulness for this case: benchmarking, data modelling, document analysis, interviews and observation, process modelling, the fishbone diagram, and the interrelationship diagram.

The outcomes of the thesis are twofold: practical and theoretical. The practical results encompass documentation of the requirements of the system as-is and how it is to be, database model, model of one of the business processes, suggestions for improvement of the database and ways of implementing the new system.

Theoretical results include the answer to the research question. All in all, the chosen techniques fared well throughout the case study as they helped identify the business needs and root causes for current problems, gather data for the practical outcomes and analyse said data. Most useful techniques for this thesis – that is to say, they were time-efficient and yielded results that gave valuable insights for the project – were interviewing, interrelationship diagram, and mock-ups.

Four main insights were gained throughout the process of writing this thesis. Firstly, business analysis techniques will yield results even in cases where there is no set goal. Furthermore, these techniques will be mostly be very useful. Secondly, suitable techniques need to be chosen. In this case, for example, benchmarking was not useful due to lack of similar

websites to KS, whereas interviews and root cause analysis techniques produced results that influenced all other work significantly. Thirdly, techniques that result in visual outputs (diagrams, images, tables, graphs) are most helpful when communicating with the client and aligning the understanding about the matter at hand. Finally, regular meetings (with the client) enable enough space for follow-up, clarifications, and keep the project in its timeframe.

When it comes to future work regarding problem analysis techniques, the following were not covered by this thesis but could be used in a case with similar features: Pareto analysis, root cause tree diagram, and focus groups¹². In addition, more research could be done into the harms of embarking on an IT project without first analysing the root problems.

¹² Focus group study was planned to be conducted within this thesis, but was eventually dropped due to it being out of the scope of the thesis.

References

- [1] International Institute of Business Analysis, A Guide to the Business Analysis Body of Knowledge, Toronto: International Institute of Business Analysis, 2015.
- [2] S. Hastie and S. Wojewoda, "Standish Group 2015 Chaos Report - Q&A with Jennifer Lynch," InfoQ, 4 October 2015. [Online]. Available: <https://www.infoq.com/articles/standish-chaos-2015>. [Accessed 2 April 2017].
- [3] S. Lauesen, Software Requirements, Harlow: Pearson Education, 2002.
- [4] E. Yourdon, Modern Structured Analysis, Prentice Hall, 1989.
- [5] B. A. Carkenord, Seven Steps to Mastering Business Analysis, Florida: J. Ross Publishing, 2009.
- [6] J. L. Whitten and L. D. Bentley, Introduction to Systems Analysis and Design, New York: Mcgraw-Hill/Irwin, 2008.
- [7] G. J. Myers, Composite/Structured Design, Van Nostrand Reinhold, 1978.
- [8] D. Cohen and B. Crabtree, "Semi-structured Interviews," July 6. [Online]. Available: <http://www.qualres.org/HomeSemi-3629.html>. [Accessed 27 February 2017].
- [9] M. Dumas, M. La Rosa, J. Mendling and H. A. Reijers, Fundamentals of Business Process Management, Springer, 2013.
- [10] N. R. Tague, The Quality Toolbox, ASQ Quality Press, 2005.
- [11] ISO/IEC/IEEE, "SE Vocab," 15 12 2010. [Online]. Available: https://pascal.computer.org/sev_display/24765-2010.pdf. [Accessed 10 March 2017].
- [12] A. Aurum and C. Wohlin, "Requirements Engineering: Setting the Context," in *Engineering and Managing Software Requirements*, Springer, 2005, pp. 1-15.
- [13] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical Software Engineering*, vol. XIV, no. 2, pp. 131-164, 2009.
- [14] Estonian Literary Museum, "Main page," [Online]. Available: <http://www.kirmus.ee/>. [Accessed 20 April 2017].
- [15] K. Metste and M. Laak, Kreutzwald. Missioon. Tegelikkus., Tartu: Eesti Kirjandusmuuseum, 2003.

- [16] C. MacDonald, "Assessing the user experience (UX) of online museum collections: Perspectives from design and museum professionals.," 1 February 2015. [Online]. Available: <http://mw2015.museumsandtheweb.com/paper/assessing-the-user-experience-ux-of-online-museum-collections-perspectives-from-design-and-museum-professionals/>. [Accessed 1 May 2017].
- [17] G. Curtis and D. Cobham, Business Information Systems, Harlow: Pearson Education, 2008.
- [18] The Chartered Institute for IT, Business Analysis, British Informatics Society, 2010.

Appendices

I. Administrator's functionalities as-is

1.1 Data management functionalities

1. The types of objects are as follows:
 - a. Events
 - b. Photos
 - c. Art
 - d. People
 - e. Stories
 - f. Topics
 - g. Books
 - h. Book pages
 - i. Book texts
 - j. Galleries
 - k. Timelines
 - l. Audio
 - m. Video
 - n. Texts
 - o. References
 - p. E-books
2. The objects have relations between them: for example, a book object is mapped to book pages; each event belongs to one timeline and so on.
3. All of the inserted objects must be displayed to the regular user.
4. The administrator must be able to
 - a. Add a new object.
 - b. Edit all of the features (such as name, date, the group that the object belongs to, and all other features that the object has).

- c. Be able to see all of the objects.
- d. Be able to search for a certain object by some of its features.
- e. Delete one or several objects at once.
- f. Group certain objects, e.g. be able to add all event-type objects that take place in Estonia into one group to be displayed in one timeline.
- g. Set the order for some objects in which the objects appear to a regular user of KS.
- h. Create relations between objects, e.g. add a photo to an event.

1.2 Administrative functionalities

- 1. The administrator must be able to insert, update and delete users.
- 2. The administrator must be able to see the existing users.

1.3 Content management

- 1. The administrator must be able to do content management on the site, that is, have the following functionalities:
 - a. Add and delete new pages and subpages
 - b. Insert, update, format and delete content (text, images, video, audio, files) on these pages
 - c. Choose which content is visible to which user groups.
 - d. Add metadata to the content
 - e. Choose which administrative users can access and change which content.

II. Functionalities of a non-administrative user as-is

1.1 General functionality

1. There must be a search function.

1.2 Timeline

1. This page contains a timeline on which events have been inserted. The events belong to different groups of events: currently the groups are European literature and people, Estonian culture and life, Estonian authors, and Kreutzwald's life and creations.
2. The legend for the colour codes for event groups must be shown.
3. Further information about the currently chosen event (one that was most recently clicked on) must be displayed. The sections for this additional information are the following:
 - a. The name of the event
 - b. The event group it belongs to
 - c. A longer description or explanation about the event
 - d. A list of other events that happened in the same year
 - i. Upon clicking to one of these events, the information about the currently chosen event must be replaced by the info about the event which was clicked on.
 - e. The year on which the event took place
 - f. Related material from the rest of the page, e.g. photos or links.
 - i. Upon clicking to one of these materials, information about that material must be shown without taking the user away from the page with the info about the current event, for example a pop-up window can be used.
4. There must be a possibility for the user to go through the events one by one in chronological order.

1.3 Graphical functionality for the timeline

1. The page must display the timeline graphically.
2. There must be year numbers (every 10th year number) visible on the timeline.
3. There user must be able to zoom in and out of the timeline.

4. There must be a possibility to go to the default view of the timeline.
5. The timeline must be scrollable in both directions (“future” and “past”).
6. The year numbers should not end in either direction, i.e. the timelines must be infinitely scrollable.
7. The events must be displayed on the timeline in the form of their short description. If the description is longer than 12 characters, the rest are omitted.
8. Each event group must have its own colour and all the events belonging to that particular group must be displayed in this colour.
9. When the user hovers over an event with their mouse, the full name of the event must be displayed.
10. Upon clicking an event, information about this event must be shown to the user as described in 1.2.3.

1.4 “Kalevipoeg” and Noor-Eesti

These two pages contain the manuscripts for “Kalevipoeg” and works by the literary grouping Noor-Eesti. Although these are realised as two different subpages of the site, their purpose and structure are the same which is why only one set of requirements is described for these two subpages. These two subpages will be referred to as “Kalevipoeg” and “Noor-Eesti”.

1. There must be an introduction to the subject (i.e. text that answers the questions “What is Kalevipoeg?” and “What was Noor-Eesti?”).
2. For “Kalevipoeg”, there must be an overview of the layout of the page, i.e. what can be found on each of its subpages.
3. Both “Kalevipoeg” and “Noor-Eesti” must have subpages for relevant texts. For “Noor-Eesti”, these texts are collections of the grouping’s work; for “Kalevipoeg”, these texts are relevant materials to the tale of “Kalevipoeg”, actual texts of the epic and a link to the “Kalevipoeg” timeline.
 - a. The “Kalevipoeg” timeline is a timeline that has the same functionalities as described for timelines above, but the only events that are displayed are those related to the epic “Kalevipoeg”.
 - b. On each subpage, there must be a link to the e-catalogue Ester for the relevant text.
4. The relevant texts are of two types.

- a. The first type of texts are displayed directly in the subpage as plain text.
 - b. The second type of texts are not displayed directly. They can be accessed in at least one of the following three different ways from the relevant subpage.
 - i. They can be downloaded as an e-book.
 - ii. They can be displayed as plain HTML text on a separate page.
 - iii. They can be displayed as scanned images of the actual text on a separate page.
5. For the texts described in 5.b.ii and 5.b.iii, the following applies:
- a. These texts must be displayed one page at a time.
 - b. The user must be able to move between these pages one by one.
 - c. A table of contents must be displayed for the user, where a list of page numbers is shown. Upon clicking a certain page number, the corresponding page must be displayed.
 - d. The user must be able to exit the viewing process and be redirected back to the subpage.
6. There must be a section on every subpage titled “Related material”, which contains links to relevant objects on the “Collections” subpage.
- a. If there are no related materials to be displayed, the entire section must not be displayed.
 - b. The related materials must be grouped by the object type, e.g. all relevant events must be displayed under the title “Events”, all people under the title “People” and so on.
 - c. After the title of the object type, the amount of this kind of relevant objects within this category must be displayed in brackets.
 - d. Only the first four relevant objects within each category must be displayed. The user must be able to click on a button which displays all of the related objects.
 - e. Upon clicking the object, the user must be redirected to the respective subpage of the “Collections” subpage.

1.5 Collections

- 1. The “Collections” subpage displays different types of objects for the viewer. These objects can be grouped into the following categories:

- a. People
 - b. Events
 - c. Photographs
 - d. Art
 - e. Galleries
 - f. Books
 - g. Articles
 - h. Audio
 - i. Videos
 - j. References
 - k. Critique
 - l. Links
2. The functionalities for the “Collections” subpage are the following:
- a. There must be an “Introduction” subpage which explains the “Collections” page and what subpages it contains and what one can expect to find from them.
 - b. For all of the aforementioned object categories, there must be a separate subpage that displays every object in the database that belongs to this category.
 - c. These subpages must display a list of the titles of each of its objects. Upon clicking the title, the user must be redirected to a new subpage or a relevant external page which contains all of the information about this specific object and materials related to it.
 - i. The functionalities described in chapter “Kalevipoeg” and Noor-Eesti point 6 must apply to this subpage as well.
 - d. If necessary, each subpage must itself have subpages that divide its objects into further groups. For example, the objects of type “People” are divided into two categories: Estonians and Europeans. The user must be displayed these subcategories as well.
 - e. The user must be able to see all of the objects in a given category at once or be displayed these objects 25 at a time with the latter being the default.
 - i. The user must be able to move between the groups of 25 one by one.
3. For book type objects, the rules outlined in chapters “Kalevipoeg” and Noor-Eesti points 4.b and 5 must apply.

1.6 Introduction

1. This subpage displays all the relevant information about the KS page itself and about the ECHA.
2. The introductions must be displayed in the following five categories:
 - a. Introduction. This subpage displays the general info about the KS, its purpose, the structure of the site and what can be found within it.
 - b. Table of contents. This subpage contains the site map of KS.
 - c. Sponsors. This subpage displays the names and, if relevant, logos of organisations that have helped the KS in any way.
 - d. Staff. This subpage contains of the people who work supply content for KS, what content they are responsible for, the sources of the content, and their contact information.
 - e. Feedback. This subpage consists of a feedback form which allows the user to send messages to the site owners.
 - a. The form must have the following fields:
 - i. Last name
 - ii. First name
 - iii. Phone number
 - iv. Email
 - v. Comments

III. Database model

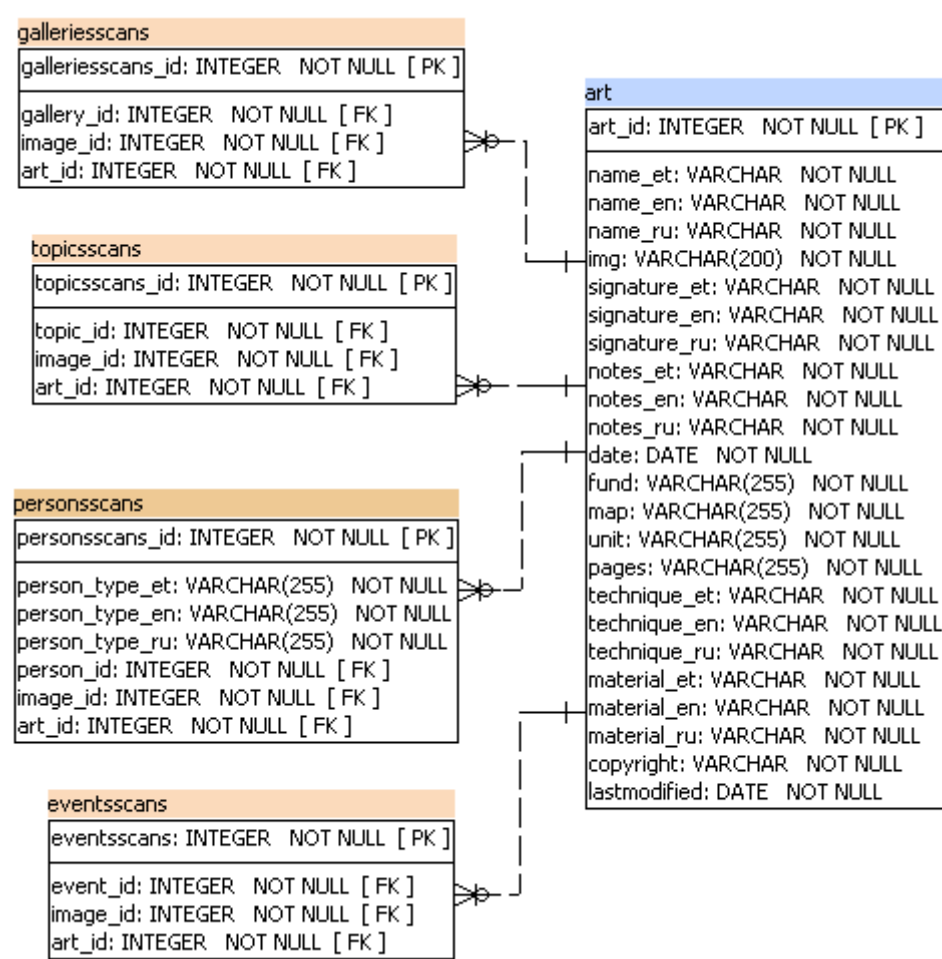


Figure 17. Table *art* and tables related to it.

erm_users	
user_id: INTEGER	NOT NULL [PK]
user_name: VARCHAR(50)	NOT NULL
firstname: VARCHAR(50)	NOT NULL
surname: VARCHAR(50)	NOT NULL
passwd: VARCHAR	NOT NULL

Figure 18. Table *erm_users*.

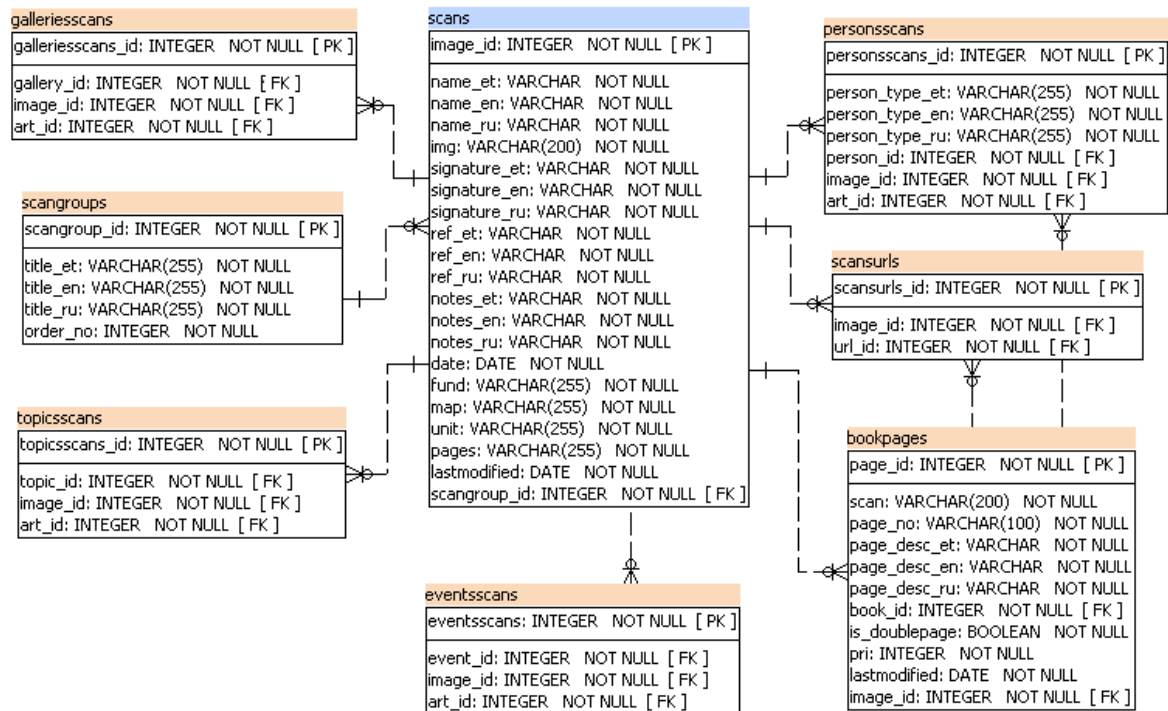


Figure 19. Table *scans* and tables related to it.

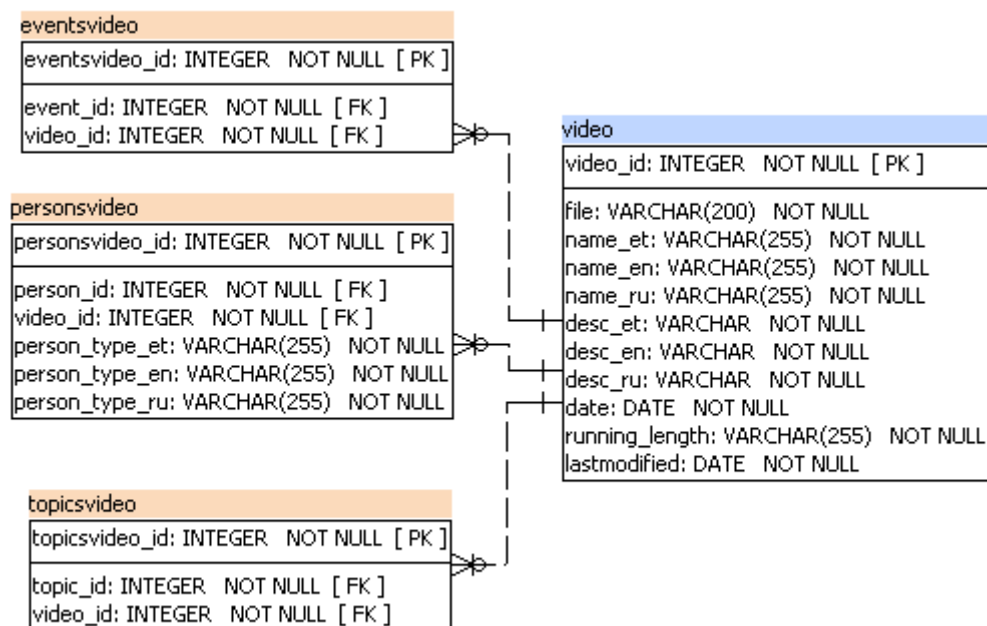


Figure 20. Table *video* and tables related to it.

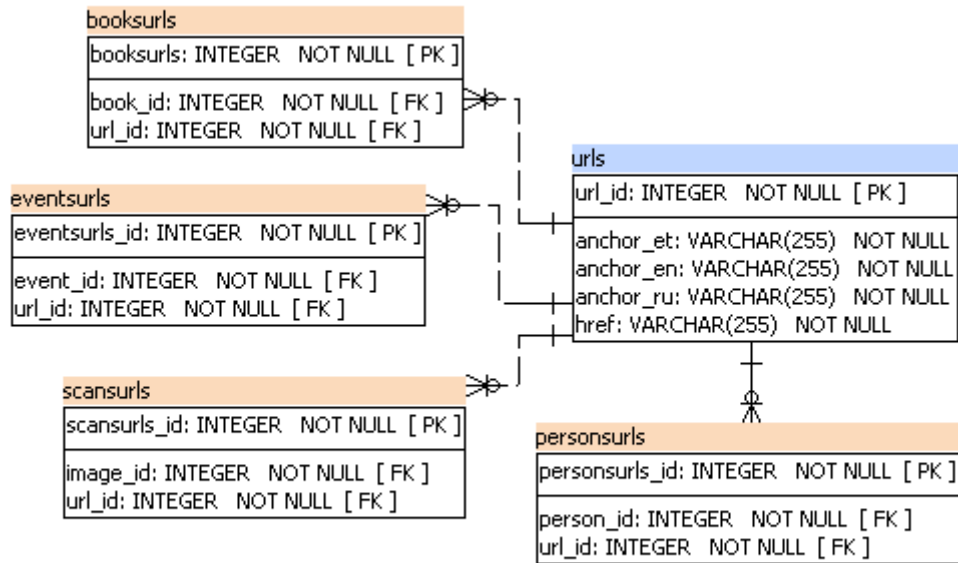


Figure 21. Table *urls* and tables related to it.

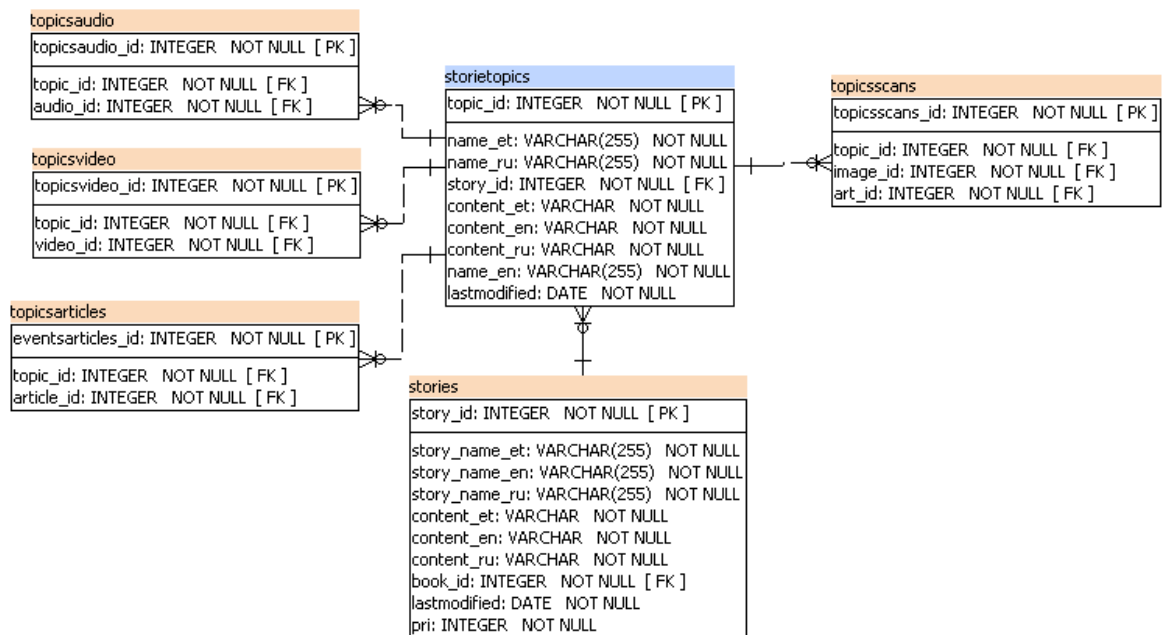


Figure 22. Table *storiетopics* and tables related to it.

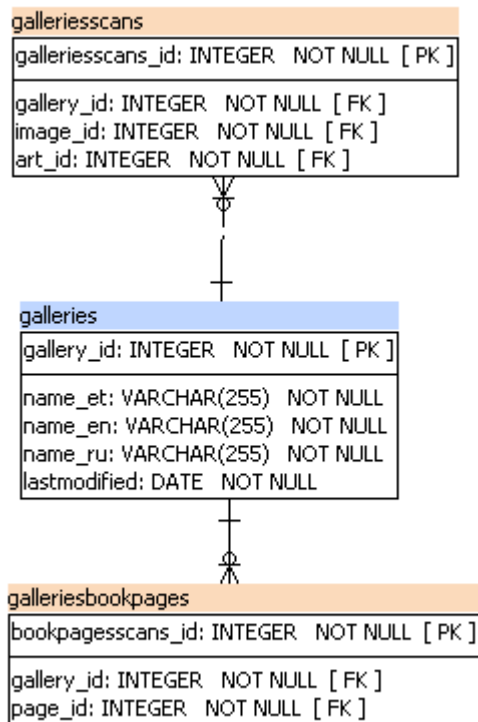


Figure 23 Table *galleries* and tables related to it.

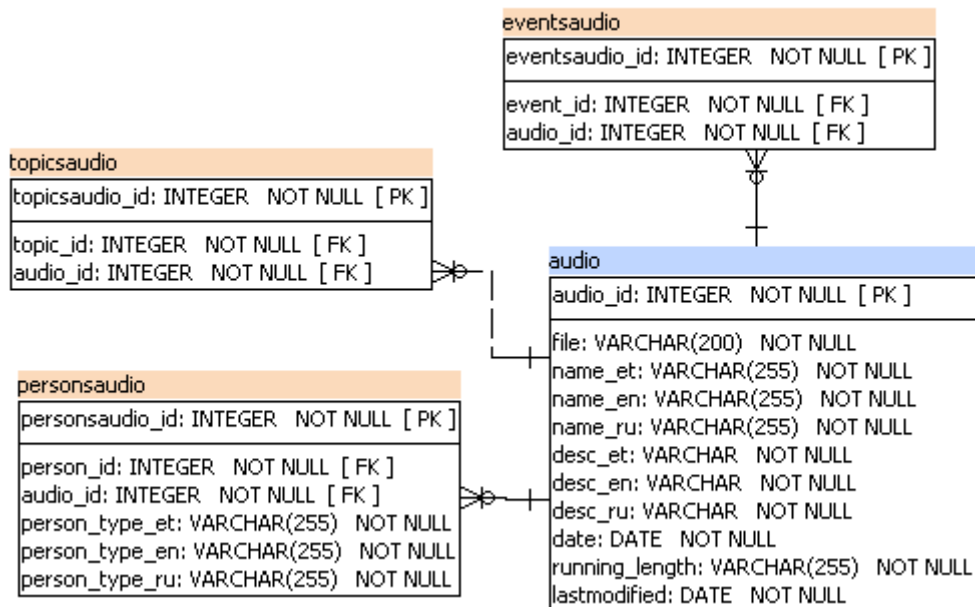


Figure 24. Table *audio* and tables related to it.

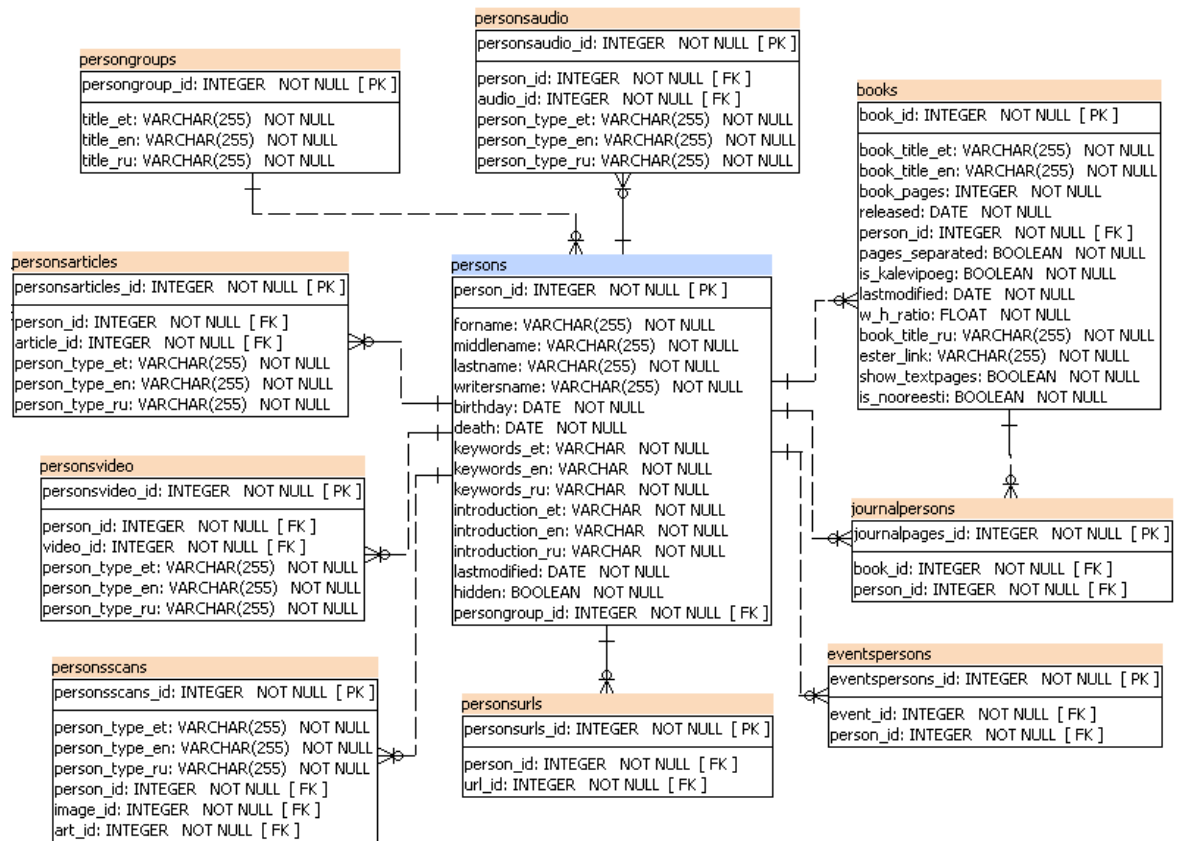


Figure 25. Table *persons* and tables related to it.

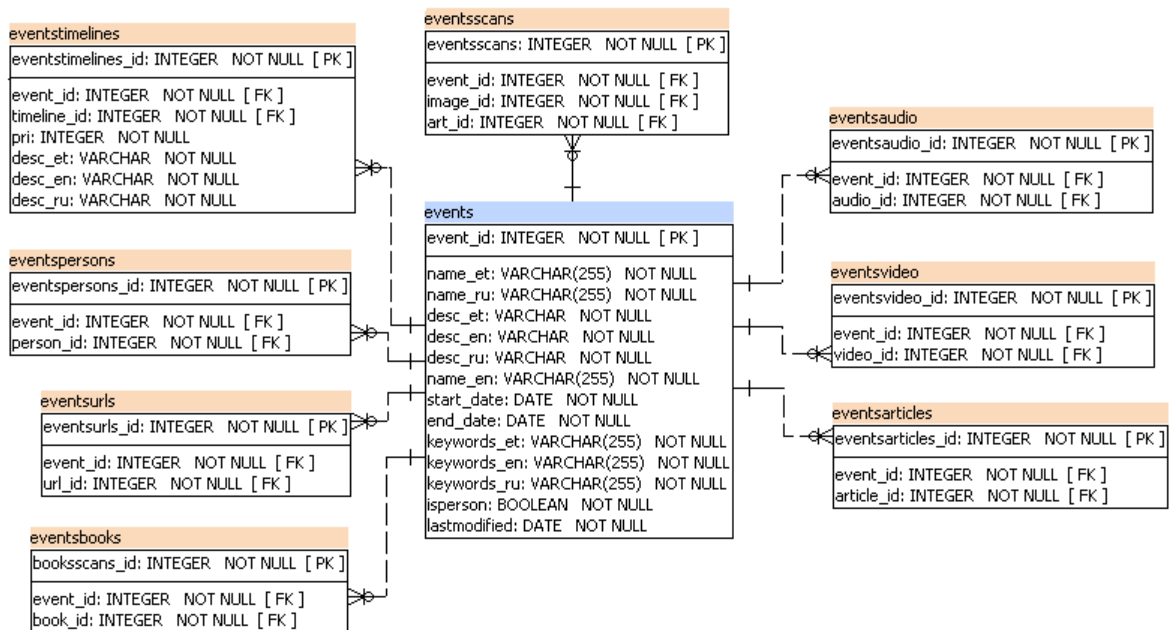


Figure 26. Table *events* and tables related to it.

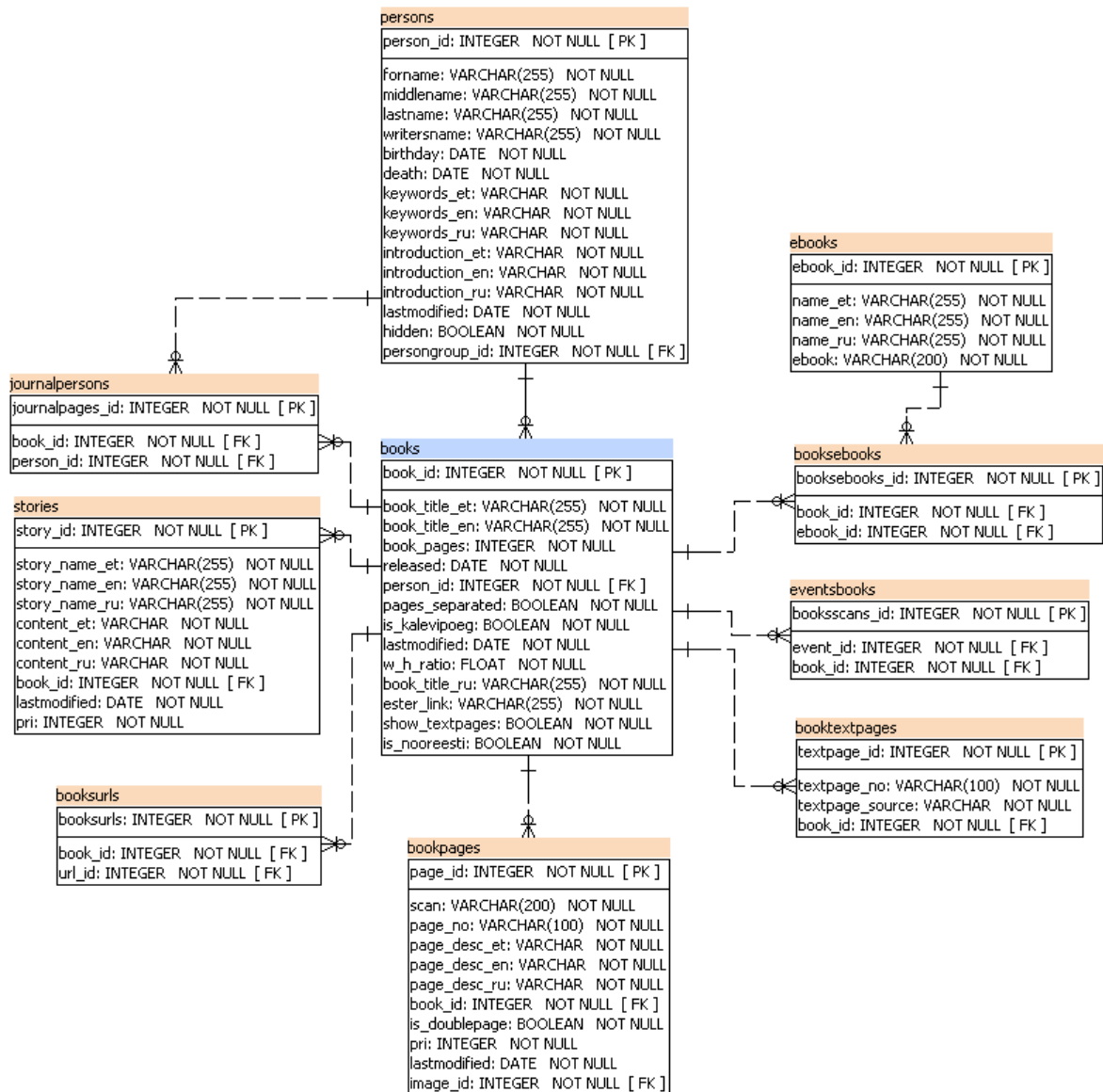


Figure 27. Table *books* and tables related to it.

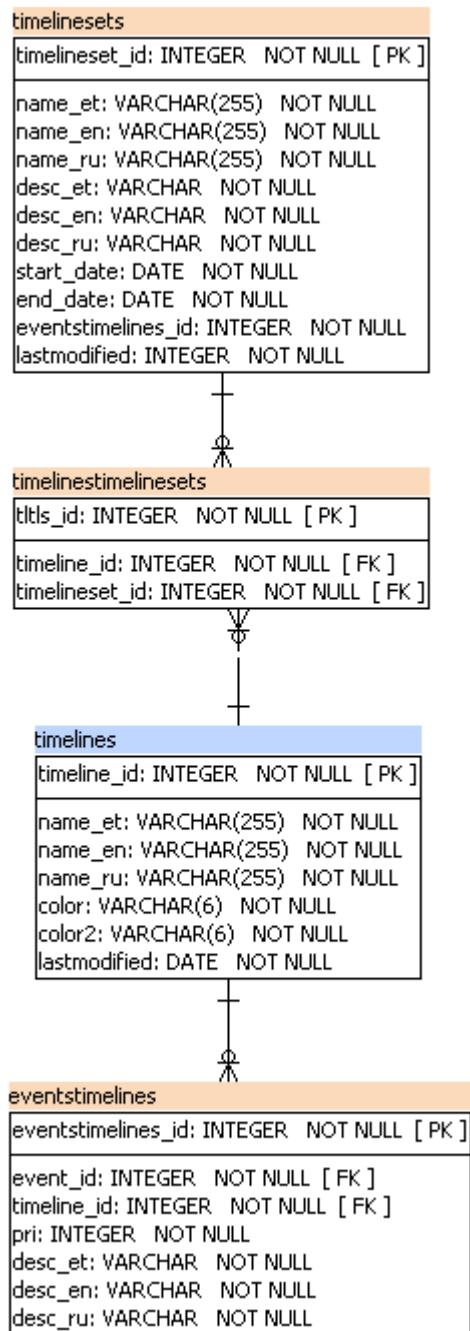


Figure 28. Table *timelines* and tables related or relevant to it.

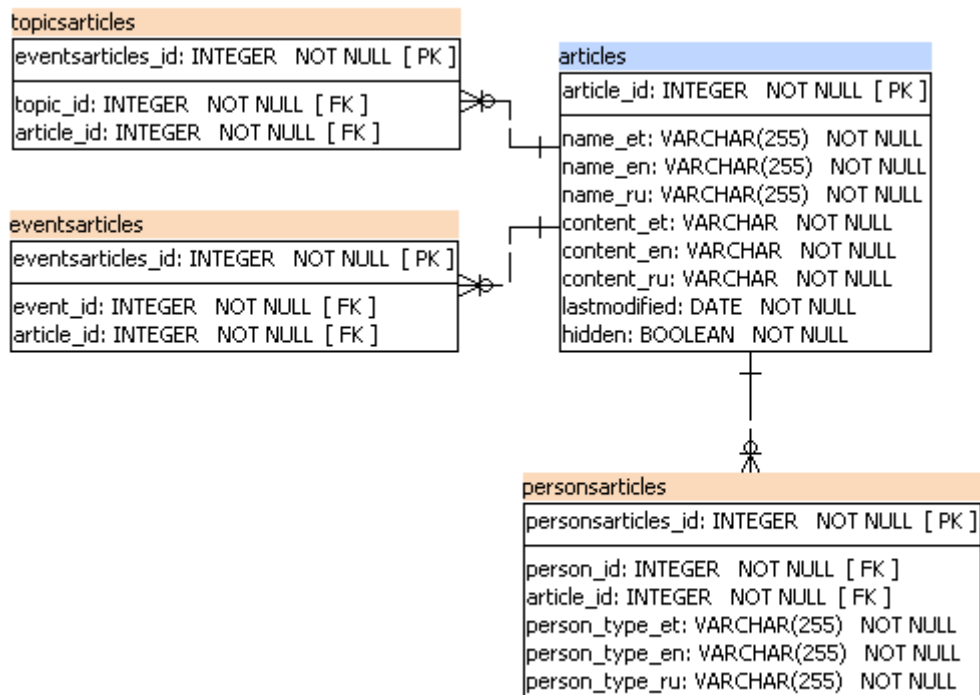


Figure 29. Table *articles* and tables related to it.

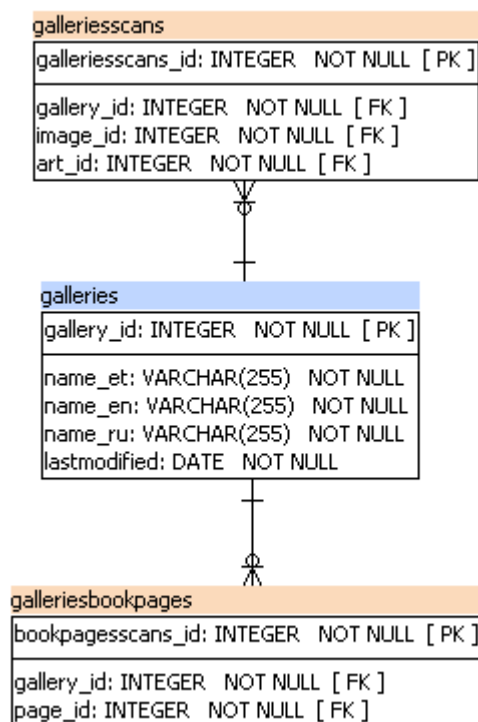


Figure 30. Table *galleries* and tables related to it.

IV. Functionalities of a non-administrative user to-be

1.1 General functionalities

1. The functionalities specified in Appendix II must be available to the non-administrative user.
2. On each object's subpage, the author of this content must be displayed.
 - a. "ECHA administrator" for content created by the administrators.
 - b. Username for content created by logged-in users (see below).

1.2 Functionalities for a logged in user

1. The user must be able to log into and out of KS.
2. The user must be able to bookmark events at the "Timelines" page. This means that there is a quick access menu on the "Timelines" page that displays the events that they have bookmarked and that they are able to navigate to with one click.
 - a. The user must be able to remove events from their bookmark list.
3. The user must be able to create relations between two objects, e.g. connect a person and a book that person has written.
 - a. The user must be able to choose whether this relation is visible to everyone, only themselves or only to people who have the link to the relation.
 - i. Seeing the content on the link does not require the viewer to be logged in.
 - ii. If it is visible to everyone, then non-logged in users must be able to see these relations on the corresponding objects' subpages under "Related materials".
4. The user must be able to create their own instances of objects: e.g. events, people, etc.
 - a. The user must be able to decide whether these objects are visible to everyone, only themselves or only to people who have the link to the object.
 - i. Seeing the content of the link does not require the viewer to be logged in.
 - ii. If they are visible to everyone, they must be displayed to the non-logged in users.

5. When the object requires a file to be uploaded, e.g. a photo, there must be a reasonable size limit applied.
6. The user must be able to create their own timelines.
 - a. The user must be able to name the timeline.
 - b. The user must be able to add events to it and delete them.
 - c. The user must be able to choose whether the timelines are visible to everyone, themselves or to anyone who has the link to the timeline.
7. The user must be able to submit proposals for change of the existing content.
 - a. An administrator must be able to see these changes and approve or reject them.
 - i. If the administrator approves the change, the information is updated.
 - ii. If the administrator rejects the change, the information is not changed.
8. The user must be able to bookmark photos in the same way as described in 3. a).
9. There is a cumulative size limit applied to the content created by each user.

1.3 Timeline

1. Each event on the timeline must have its own URL.
2. There must be a search functionality within this subpage on the events.
3. The user must be able to choose which event groups they wish to view.
4. The duration of events must be indicated on the timeline.

1.4 Collections

1. The user must be able to view up to four photos at once, side by side.
 - a. The user must be able to choose the photos they want to view together.
2. When viewing a photo, there must be buttons that display the next and the previous photo without needing to exit the interface for viewing photos.
3. The user must be able to zoom into and out of a specific place on the photo.
4. The photos that the user viewed during the same session must be marked in a way that helps the user differentiate between photos they have already seen and those they have not.
5. When a photo originates from another source, the source must be displayed to the user, for example as a link.
6. The user must not be able to download the photos.

7. The user must be able to search for specific objects under that object type's subpage:
for example for photos only from the 1950s.
8. The user must be able to view two texts at once in a split-screen mode in order to
compare them.

V. Administrator's functionalities to-be

1. The functionalities presented in Appendix I must be implemented
2. The administrator must be able to create new object types, define its attributes and relations to other object types.
3. Each relation between two object types must have a description.
 - a. For example “written by”, “parody of”, “participates in”, etc.
4. The administrator must be able to add content from other databases via KS.
 - a. The databases supported must be the following:
 - i. MUIS (Information System of the Estonian Museums)
 - ii. FOTIS (Film Archives of the National Archives of Estonia)
 - iii. ESTER (Online catalogue of Estonian libraries)
 - iv. KIVIKE (Estonian Literary Museum's Archives)
 - v. SAAGA (Estonian National Archives' online database)
 - b. When content from other databases is added, the source of this content must be displayed to the user.
 - i. This source must be saved to the database automatically.
5. The administrator must be able to create and delete relation types between object types and define its attributes.
6. People-type objects must be automatically added to event-type objects in order to display them on timelines.

VI. List of Problems

1.1 Data model

1. One book can have only one author.
2. There is no object type corresponding to manuscripts, letters, or magazines.
3. Relations between two object instances need to be created manually, especially since some objects, e.g. photos, can be inserted automatically with a script in a bulk.
4. Some objects need to be duplicated: for example, if a book or a person needs to be present on a timeline, a separate event-type object needs to be created by hand.
5. When adding photos, they need to be uploaded to two separate databases: Kivike and KS.
6. Some events and relations that have been inserted into the database are not displayed to the user on the timeline.
7. When adding a relation between two objects, there is functionality to change the relation even before it has been created during choosing the object that the active object is too be related to.

1.2 Administrators' page

1. When searching in the admin page, for example when creating a relation between a book and a photo, then it is difficult to differentiate between search results because
 - a. The results only display one attribute of the object, e.g. description.
 - b. This attribute can be the same for several objects, e.g. auto portraits of a person will all be described with the name of that person.
 - c. For photos, no image is displayed, only the description.
2. The “Select all” option will only select the objects on the current page (50 objects). Note that there are, for example, 220 pages of photos.
3. The administrator cannot choose how many object per page they wish to display.
4. The administrator is shown all of the page numbers at once (see Figure 31).
5. The current content management system is too complicated to be understood and used – it displays the website hierarchically with folders and files. Creating a vertical side menu or other page elements is out of the scope of the administrators.

6. The current content management system's WYSIWYG text insertion and formatting does not work in Google Chrome.

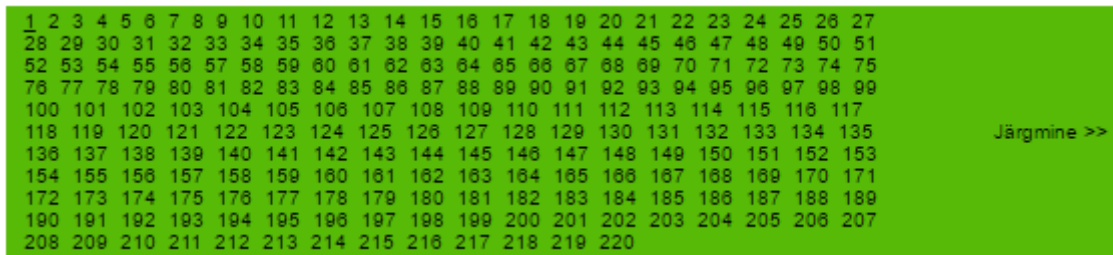


Figure 31. How the page numbers on the administrators' site are displayed.

1.3 KS website

1. There is no “Help” (either as a separate subpage or intermittently throughout the page) or “Frequently asked questions” section.
2. Search function displays results in a non-understandable manner.
3. The timeline display is cluttered.
 - a. Especially since for some events, only the year is known, in which case the date is saved as January 1st 00:00, meaning all of these events will be displayed at exactly the same point on the timeline.
4. The only way to find one specific event on the timeline is to go through them one by one.
5. The duration of an event is not displayed graphically on the timeline.
6. The section that displays events that happened in the same year as the currently active event on the “Timelines” page can be too long compared to the other contents on the page (see year 1922 on the page, for example, where all 94 other events that happened on this year are displayed as a bullet-pointed list on every event’s introduction page).
7. The user cannot browse through photos one by one.
8. There are no *forward* and *back* buttons for viewing photos.
9. Icons of photos sometimes “cut off” half of a person’s head on the image.
10. The subpage “Collections” – “Photos” does not display the photos themselves as a list, just their descriptions.
11. The interface for reading a book in HTML is not always responsive:
 - a. Sometimes a part of the text is cut off by the edge of the browser window without a scrollbar.

- b. The text displayed is not responsive with certain dimensions of the browser window.
- 12. When displaying photos, sometimes the error message “The requested content can not be loaded” is displayed disproportionately often.
- 13. When viewing a book in the HTML format or as scanned images in the special interface, the table of contents is just a list of page numbers.
- 14. When closing the interface for viewing books, it takes the user to the respective object subpage on the “Collections” menu, even when the user opened the interface from somewhere else.
- 15. The “Related materials” section sometimes is several times longer than the actual contents of an object’s subpage, meaning there is a lot of useless whitespace.

VII. Structure of subpages

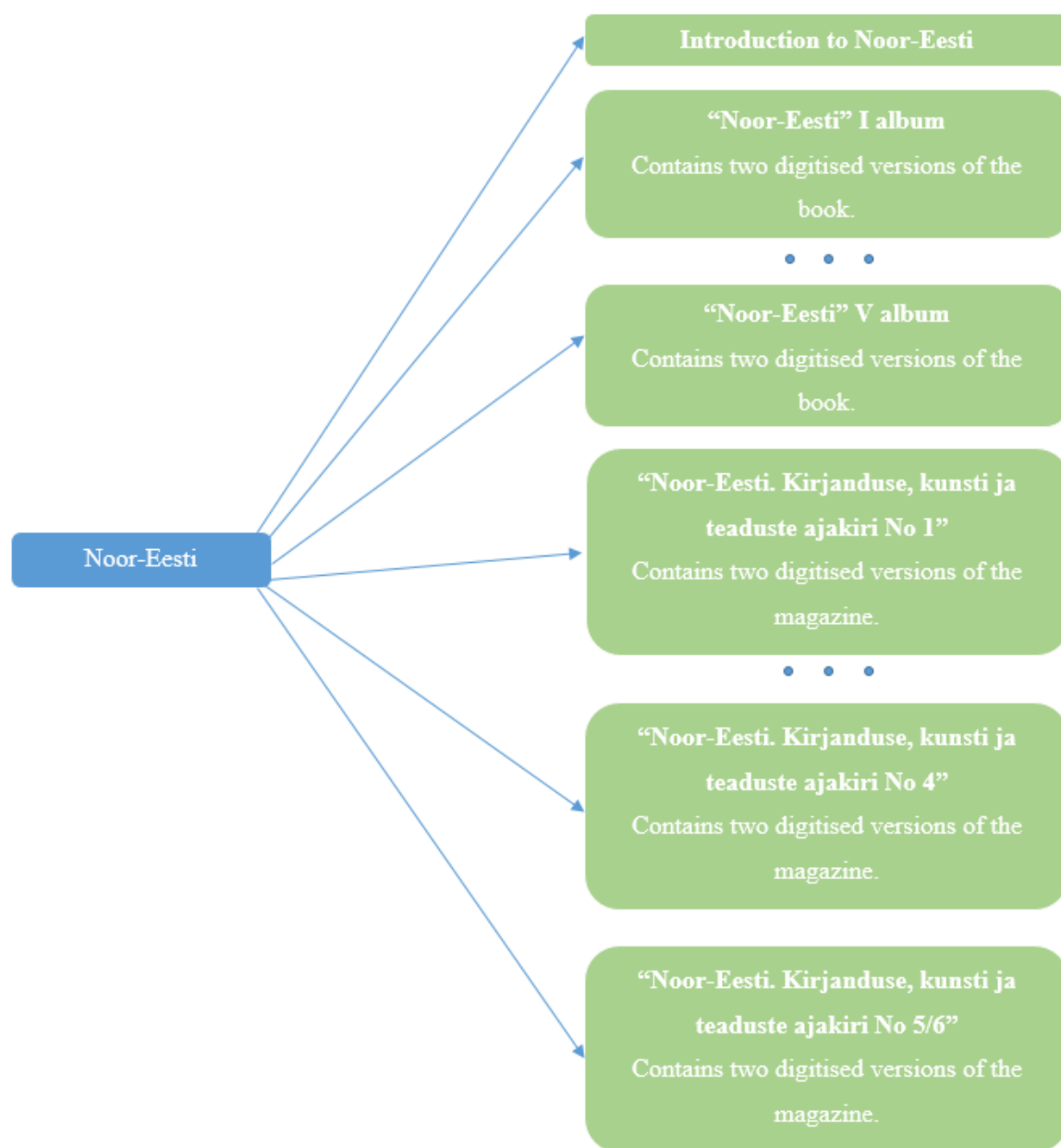


Figure 32. Layout of the page "Noor-Eesti"

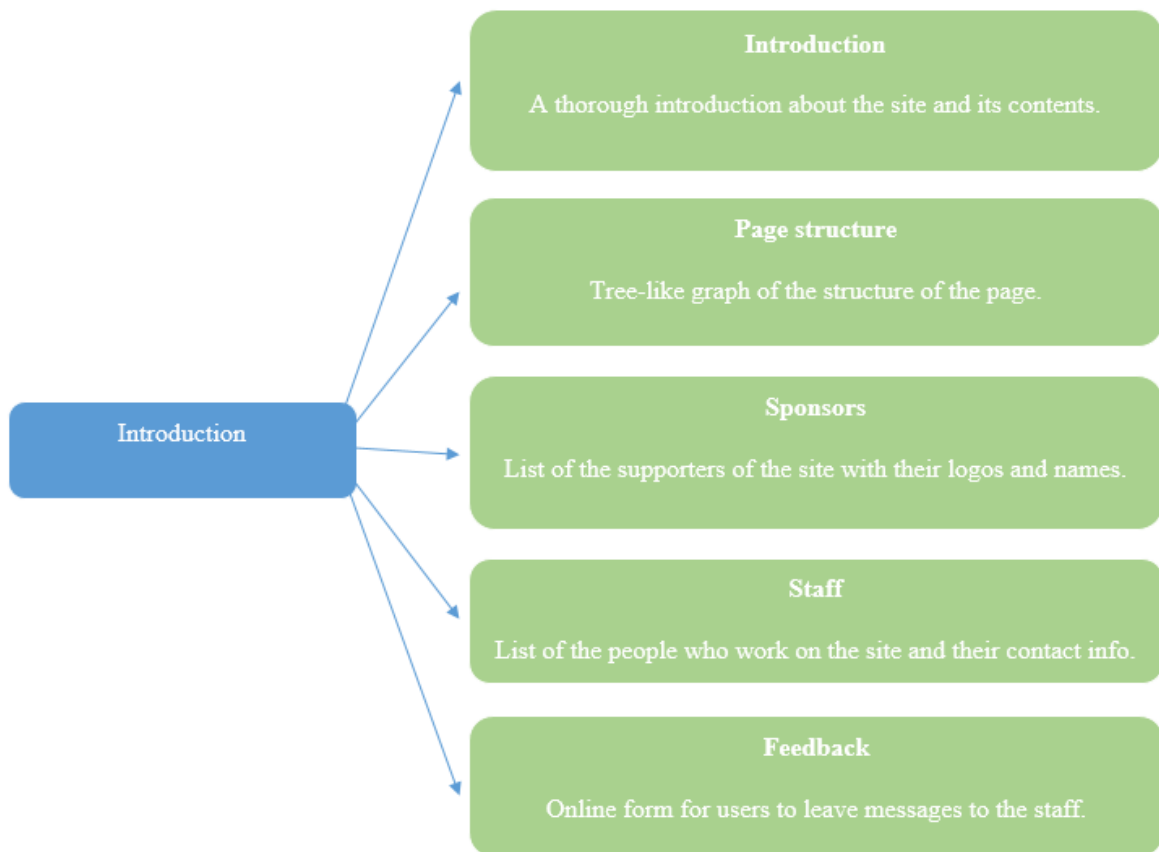


Figure 33. Layout of the page "Introduction".

VIII. Appearance of KS

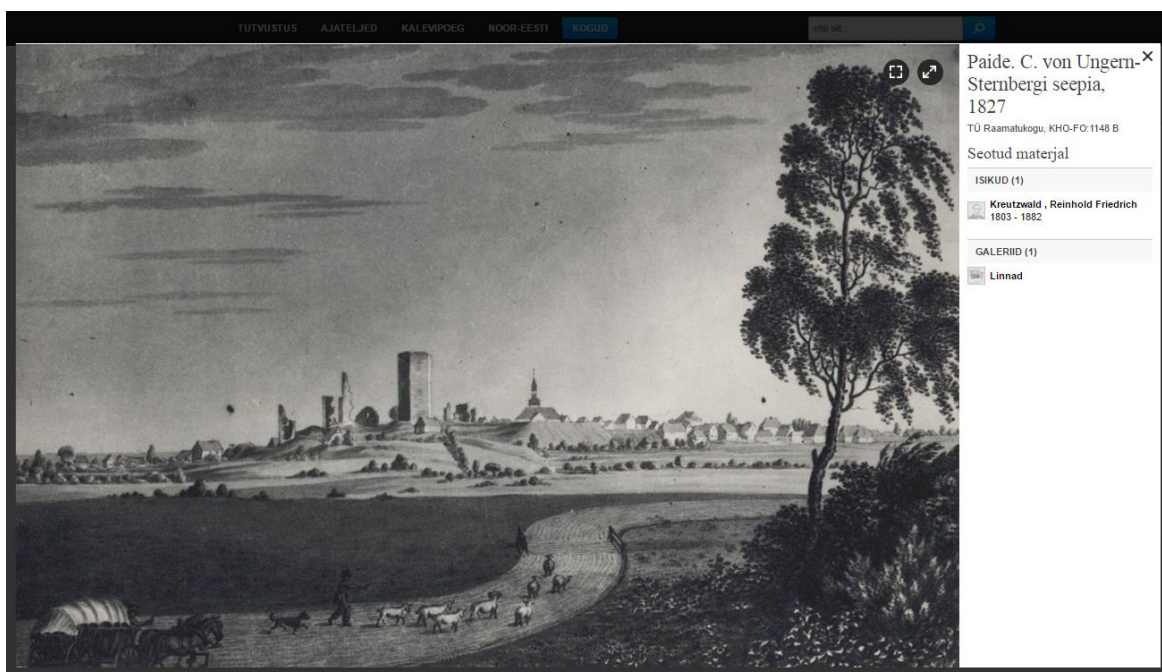


Figure 34. Interface for viewing photos.

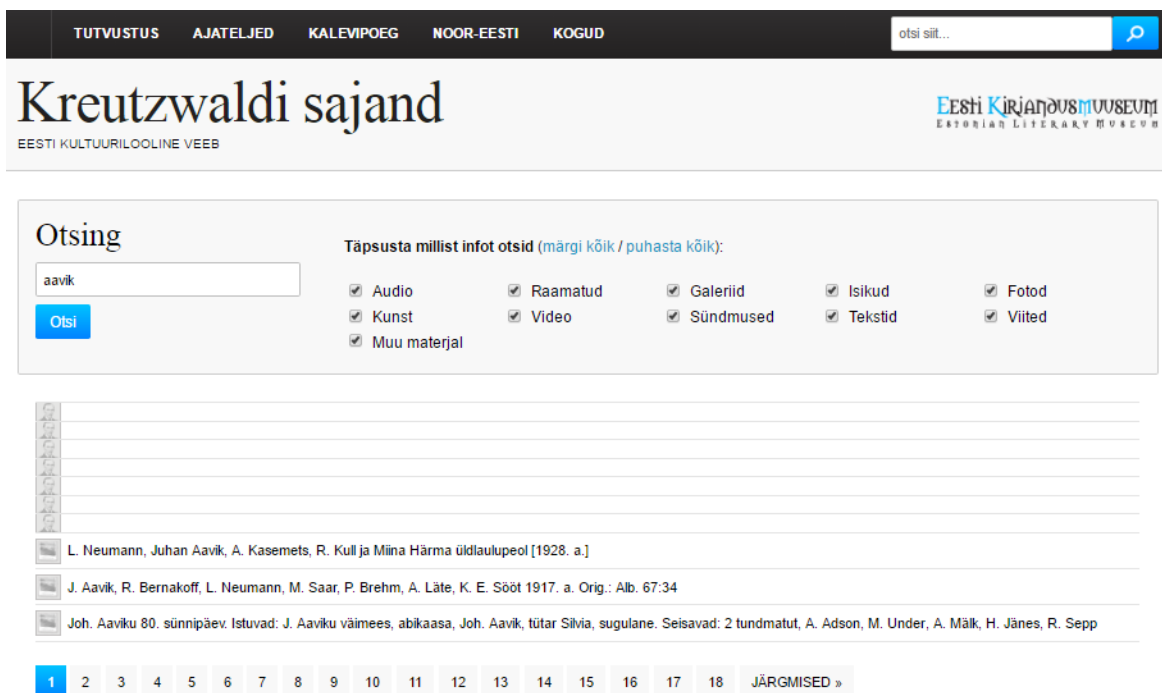


Figure 35. The search function.



Figure 36. Interface for displaying a book in HTML format.

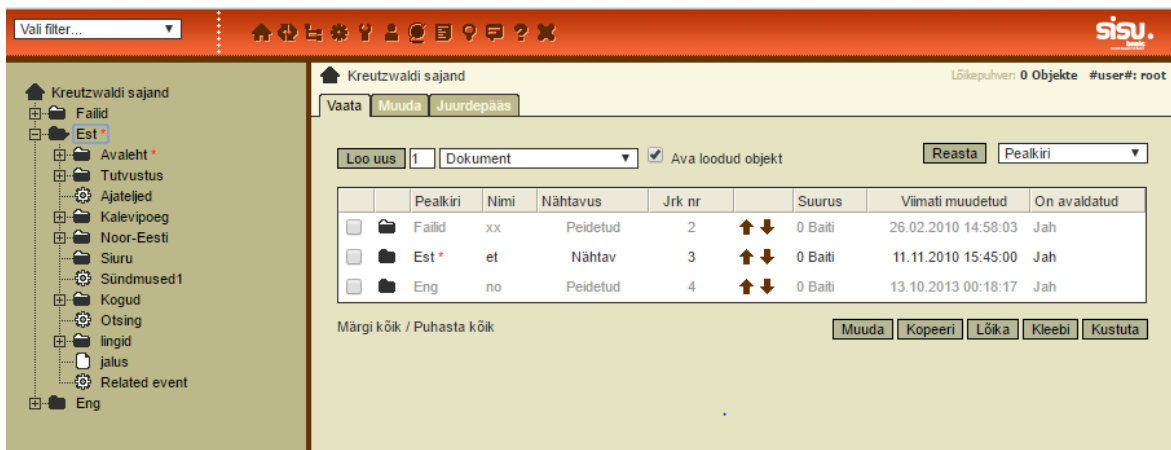


Figure 37. Appearance of the content management site for the administrators.

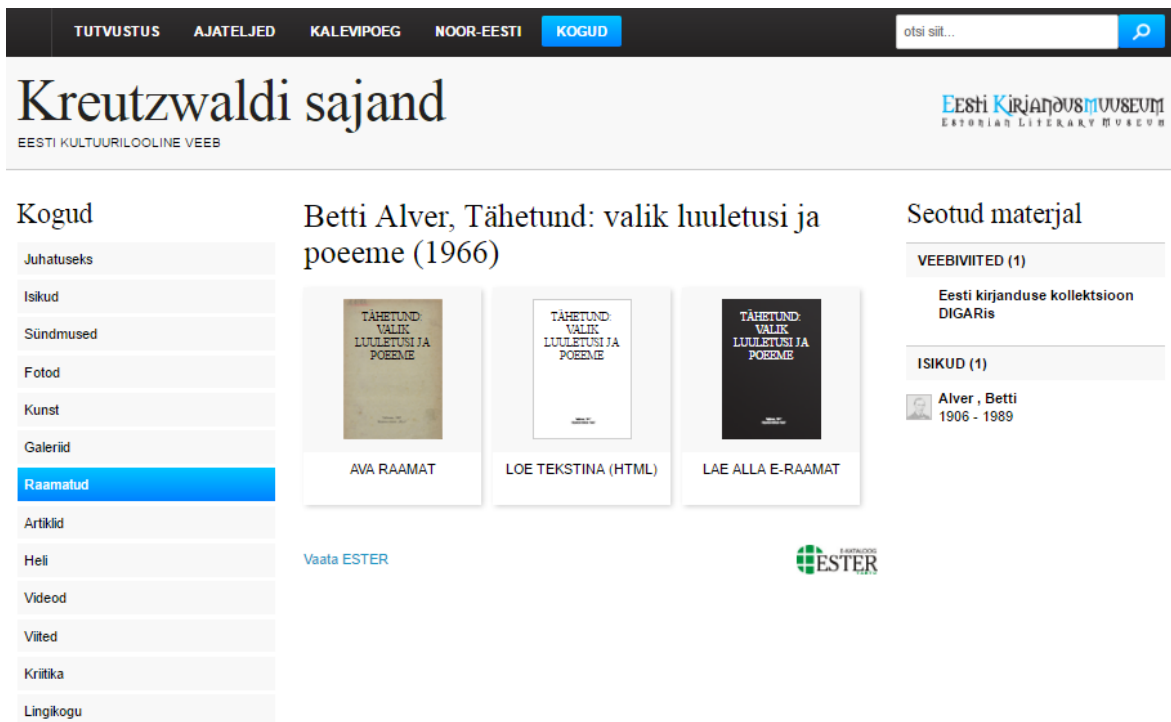


Figure 38. The subpage of a book-type object.

IX. Proposed changes to the data model

Table name	Attributes	Relations to other tables
<i>manuscripts</i>	Integer <i>manuscript_id</i> Varchar <i>manuscript_title_et</i> Varchar <i>manuscript_title_en</i> Varchar <i>manuscript_title_ru</i> Integer <i>manuscript_pages</i> Integer <i>pages_separated</i> Boolean <i>is_kalevipoeg</i> Boolean <i>is_nooreesti</i> Date <i>lastmodified</i> Float <i>w_h_ratio</i> Boolean <i>show_textpages</i>	1:m for <i>manuscripts</i> - <i>manuscriptspersons</i> 1:m for <i>manuscripts</i> - <i>booktextpages</i> 1:m for <i>manuscripts</i> - <i>bookpages</i> 1:m for <i>manuscripts</i> - <i>stories</i> 1:m for <i>manuscripts</i> - <i>manuscriptsevents</i> 1:m for <i>manuscripts</i> - <i>manuscriptsurls</i> 1:m for <i>manuscripts</i> - <i>manuscriptsebooks</i>
<i>manuscriptspersons</i>	Integer <i>manuscriptspersons_id</i> Integer <i>person_id</i> Integer <i>manuscript_id</i>	1:m for <i>persons</i> - <i>manuscriptspersons</i> 1:m for <i>manuscripts</i> - <i>manuscriptspersons</i>
<i>manuscriptsevents</i>	Integer <i>manuscriptsevents_id</i> Integer <i>event_id</i> Integer <i>manuscript_id</i>	1:m for <i>events</i> - <i>manuscriptsevents</i> 1:m for <i>manuscripts</i> - <i>manuscriptspersons</i>
<i>manuscriptsurls</i>	Integer <i>manuscriptsurls_id</i> Integer <i>url_id</i> Integer <i>manuscript_id</i>	1:m for <i>urls</i> - <i>manuscriptsurls</i> 1:m for <i>manuscripts</i> - <i>manuscriptsurls</i>
<i>manuscriptsebooks</i>	Integer <i>manuscriptsebooks_id</i> Integer <i>ebook_id</i> Integer <i>manuscript_id</i>	1:m for <i>ebooks</i> - <i>manuscriptsebooks</i> 1:m for <i>manuscripts</i> - <i>manuscriptsebooks</i>
<i>letters</i>	Integer <i>letter_id</i> Varchar <i>content</i> Integer <i>sender_id</i> Integer <i>receiver_id</i> Date <i>date_sent</i>	1:m for <i>persons-letters</i>

	Date <i>lastmodified</i>	
<i>lettersscans</i>	Integer <i>lettersscans_id</i> Integer <i>letter_id</i> Integer <i>scan_id</i>	1:m for <i>letters</i> – <i>lettersscans</i> 1:m for <i>scans</i> - <i>lettersscans</i>
<i>personsbooks</i>	Integer <i>personsbooks_id</i> Integer <i>person_id</i> Integer <i>book_id</i>	1:m for <i>persons</i> - <i>personsbooks</i> 1:m for <i>books</i> - <i>personsbooks</i>

Figure 39. Tables to be added to the data model as per author's suggestions.

X. Mock-ups of the system to be

BOOKS

ADD BOOK

123...255

"Heading of a Book"	Author's Name	Published on	DETAILS	RELATIONS
"Heading of a Book"	Author's Name	Published on	DETAILS	RELATIONS
"Heading of a Book"	Author's Name	Published on	DETAILS	RELATIONS
"Heading of a Book"	Author's Name	Published on	DETAILS	RELATIONS

25

 items on a page

Figure 40. The starting point for administrators for creating a relation between two objects.

RELATIONS OF "HEADING OF A BOOK"

Person: Author's Name	Author of	DETAILS	DELETE RELATION
Event: Publishing of the book (1845)	Publishing of	DETAILS	DELETE RELATION

ADD RELATION

Figure 41. The view after clicking “Relations” of an object for the administrator when creating a relation between two objects.

CHOOSE OBJECT TYPE

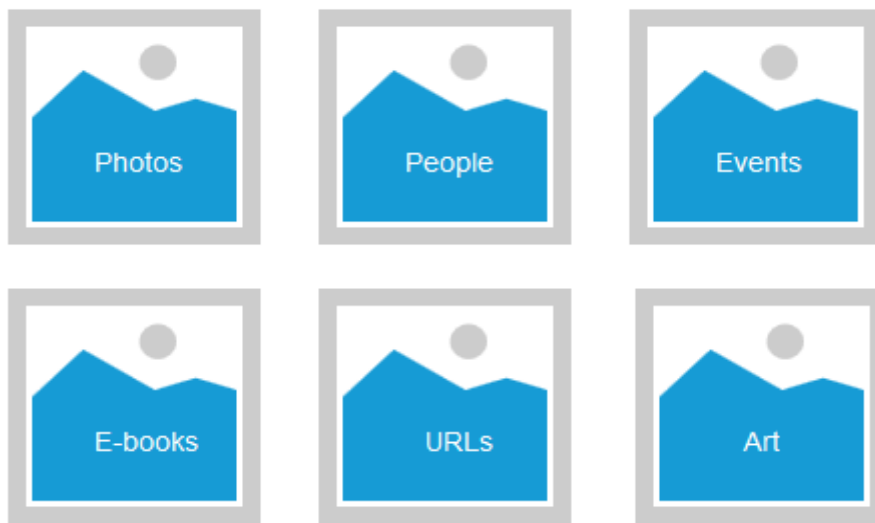


Figure 42. The view after clicking "Add relation" on the previous view for an administrator when creating relations.

CHOOSE OBJECT

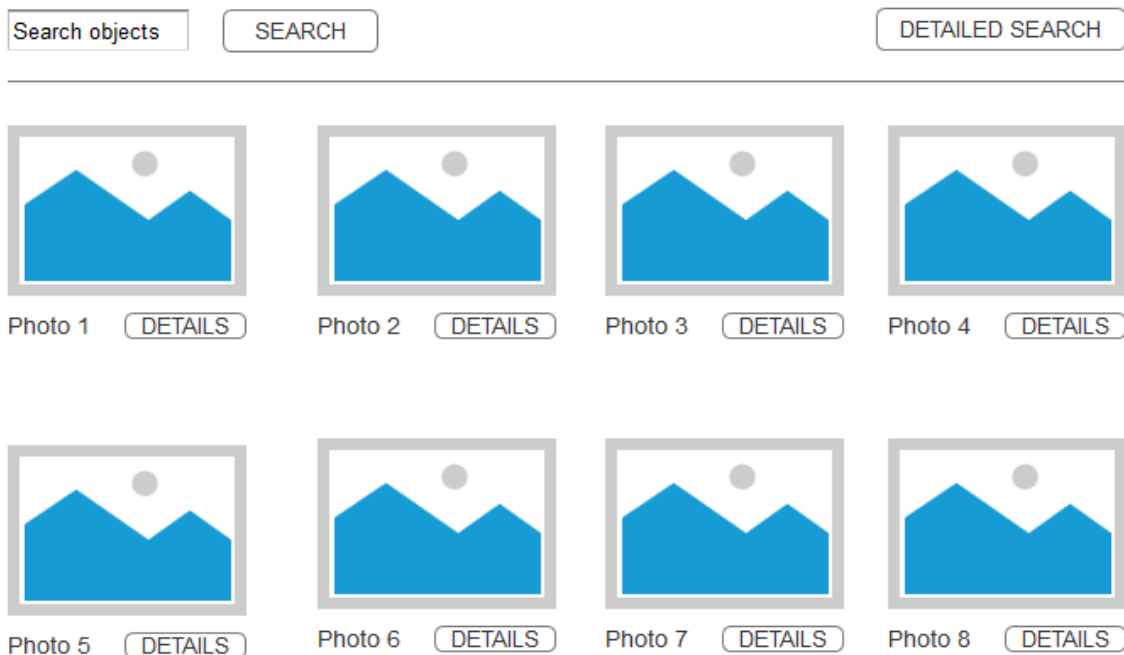


Figure 43. The view for the administrator after choosing object type "Photo" in the previous view.

CHOOSE OBJECT

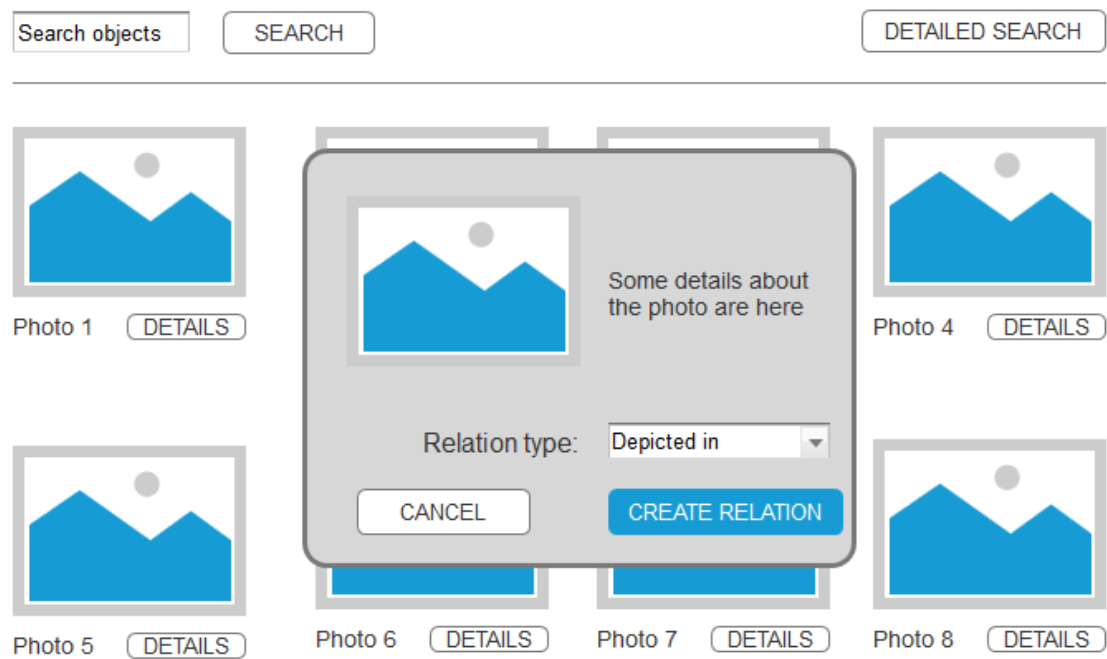


Figure 44. The final view before confirming a relation and after choosing a specific object.

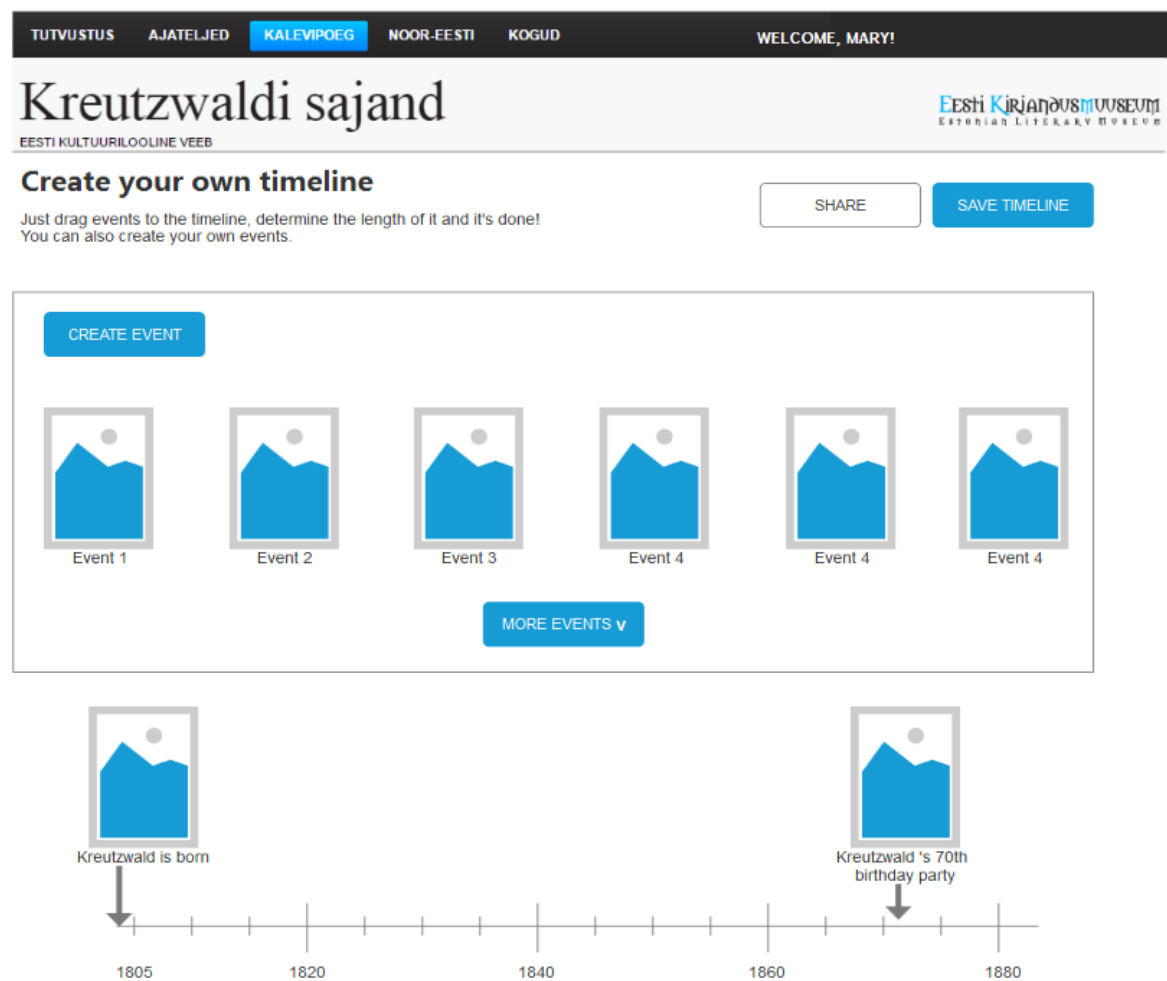


Figure 45. Mock-up of logged in user timeline creation.

Licence

Non-exclusive licence to reproduce this thesis and make the thesis public

I, Tuule Sõber,

1. hereby grant the University of Tartu a free permit (non-exclusive licence) to
 - 1.1. reproduce, for the purpose of preservation and making it available to the public, including uploading to the DSpace digital archives until expiry of the term of validity of the copyright, and to
 - 1.2. make available to the public via the online environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright,

my thesis

Aligning Business Needs with the IT Solution Design for the Historic Cultural Site Kreutzwaldi Sajand,

Supervised by Fredrik Payman Milani,

2. I am aware that the author retains these rights.
3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, 10/05/2017