Requirements specification

“Visit to the doctor”
Table of Contents

1. Introduction ............................................................................................................................. 3
   1.1. Purpose ............................................................................................................................... 3
   1.2. Scope ................................................................................................................................. 4
   1.3. Definitions, Acronyms and Abbreviations ...................................................................... 4
   1.4 Overview .............................................................................................................................. 5

2. Overall Description .................................................................................................................... 6
   2.1. Product perspective ............................................................................................................. 6
   2.2. User characteristics .......................................................................................................... 7
   2.3. Product functions .............................................................................................................. 9
       2.3.1 Major use cases .......................................................................................................... 9
       2.3.2 UC2 refinement ........................................................................................................... 16
   2.4. Expectations ...................................................................................................................... 20
       2.4.1. System expectations ................................................................................................ 20
       2.4.2. Software expectations ............................................................................................... 21

3. Specific Requirements ............................................................................................................... 22
   3.1 External Interface Requirements ..................................................................................... 22
   3.2 System Requirements ....................................................................................................... 22
   3.3 Functional Software Requirements .................................................................................. 23
   3.4 Non-Functional Software Requirements ......................................................................... 25

4. Requirements Prioritization .................................................................................................... 27
   4.1 Requirements hierarchy ..................................................................................................... 27
   4.2 Requirements prioritization ............................................................................................... 29
   4.3 High and medium priority goal modelling ..................................................................... 30

5. Requirements Traceability ....................................................................................................... 33
   5.1 Requirements traceability model ...................................................................................... 33
   5.2 Requirements traceability towards goals ......................................................................... 34
   5.3 Functional requirements traceability towards Use Cases ............................................... 36

Revision list ............................................................................................................................ 37
Reference list ............................................................................................................................ 39
Used diagram resources .......................................................................................................... 39
1. Introduction

Following document describes a new patient-doctor portal “DocVisit”, which allows to book time online for a family doctor appointment and to store family doctor related medical history in one place. Development of the family doctor system and increasing the role of the family doctor and the family nurse in our healthcare system is an important strategic objective of the Health Insurance Fund (Estonian Health Insurance Fund, 2018).

1.1. Purpose

This requirements specification document describes the functions and requirements for a new patient portal called the DocVisit. Both main and supportive objectives are in accordance with the current strategy of Estonian Health Insurance Fund.

The main objective is:

1. To make the time booking process for family doctor’s appointment more convenient by enabling real-time overview of free time slots and online booking possibilities.

Supportive objectives are:

1. To store all the relevant information for the patient in one system;
2. To enable patients to view and change their appointment times;
3. To allow patients to prepay visit fees if needed (mainly for patients who have no health insurance, but also for services not covered by health insurance);
4. To provide patients a view of their medical history from family doctor and family nurse visits and referral letters;
5. To create a calendar of reminders about health related actions. For example, schedule vaccination dates, follow-up appointments, health certificates renewals etc.

Currently the information about free slots for family doctor appointments is mainly available via phone. DocVisit will enable better schedule planning and finding suitable slots, as the information would be accessible via web.

New patient portal would be designed for both – patients and medical staff (family doctors and nurses). The system will include all family doctors who are active in Estonia, but not special care doctors.
1.2. Scope

New portal will allow patients to book/change time to their family doctor online. This will solve long call que problems which is an important issue for most family doctors. Portal will also contain patient’s medical history; including test results, vaccinations, previous doctor visits and other information that family doctor wants to share with patient. Having this information online and available to patient will help to solve family doctors’ and nurses’ work overload problem, because there will be less phone calls from the patients that nurse or doctor has to answer to (asking test results, vaccination deadlines etc). New portal will contain all active family doctors in Estonia, so all patients could be able to use it. New portal will enable doctors to send reminders to patients about upcoming vaccination and other visit deadlines. This will improve healthcare quality and save time for both doctors and patients.

New portal will be web-based online application that is available on all major browsers. Portal should be user friendly in all usual devices (PC, mobile, tablet).

New portal will not provide possibility to book time to any other medical specialists besides family doctor. Portal will not have any other medical history that is not connected to family doctor (for example dental medical history, visits to other specialty doctors). There will be no special app created for mobile devices. Geographical scope of the project is Estonia; other countries might be considered in the future if there will be demand.

1.3. Definitions, Acronyms and Abbreviations

Throughout the document following terminology and abbreviations are used:

“DocVisit” - Name of the new patient/doctor portal. Following specifications describe main functionalities of this portal.

“System” - In this context meant as DocVisit portal.

“Appointment” – In this context means a certain time for patient visit to doctor’s office that is booked through DocVisit system. Appointment exists in the system from the moment it is booked by patient and until it is cancelled or realised in time.

“Doctor” as Family Doctor - First contact of a person with health concerns in the healthcare system. In collaboration with a family nurse diagnoses and treats most diseases, monitors the child’s progress and persons with chronic conditions, performs minor surgical procedures, refers patients to examinations and analyses, vaccinates, binds the wounds, removes stitches and, if necessary, makes
home visits, gives advice on care, injuries or intoxications, and in terms of preventive measures, advises all persons included on his or her list (Estonian Health Insurance Fund, 2018).

“Health Insurance Fund” (HIF) – Estonian Health Insurance Fund

“Nurse” as Family Nurse - Has independent appointments where he or she teaches, advises and guides people in promoting and maintaining health and preventing disease. The family nurse provides assistance in the event of illness and organizes medical records, in addition, the role of a nurse is to monitor people with chronic illnesses. If necessary, the family nurse consults with the family doctor or refers the person to the appointment of the family doctor. Everyone has the right to choose their own family doctor (Estonian Health Insurance Fund, 2018).

“Medical Specialist” - Doctor specialised to a certain medical field.

“Medical Staff” – an actor in DocVisit system combining both Doctor and Nurse roles

“Referral” as Referral Letter - Document with a referral to a medical specialist posed by Doctor

“Patient” - Person who is included to the family doctor list.

“Representative” - In the context of the current document a person who has a legal right and a duty to take care of another person. Parents of the underage children also fall into this category.

“Represented person” - In the context of the current document a person who, on account of the youth or defect of understanding, or for some other cause, is unable to execute her rights.

“User” - in this context all portal users, who can be patient, doctor, nurse or representative. This does not include system administrators.

1.4 Overview

Following document gives overall description and expectations regarding the system and contains a list of functional and non-functional requirements.

Product overall description can be found in part 2. Part 2.1 gives short overview of the product. Part 2.2 describes user characteristics and relationship dependencies between main actors. Part 2.3 shows Use Case diagram and includes Use Case templates that cover Medical Staff and Patient interactions with the system. In part 2.3.1 UC2 was chosen as the most important activity of the system and was used for further refinement showing its main classes, relations and sequence of activities. Part 2.4 is about main expectations for the system.
Specific requirements can be found from Table 1 in part 3. This part is divided into 4 blocks: external interface requirements (part 3.1), system requirements (part 3.2), functional software requirements (part 3.3) and non-functional software requirements (part 3.4).

Requirements prioritization is described in part 4. In part 4.1 system main goal is divided to sub-goals and requirements are grouped into sets based on the sub-goals. As a result requirements hierarchy is shown in Graph 3. Part 4.2 describes requirements prioritization process based on Analytic Hierarchy Process (AHP) method. Prioritization is made on sub-goal level and results can be found in part 4.2. Based on conducted prioritization high and medium priority goals are refined into smaller steps. Created new goal hierarchy can be found in part 4.3.

Requirements traceability is described in part 5. First model for traceability is described shortly and then it is followed by matrix in part 5.1. Part 5.2 contains whole requirements list (Table 2) updated with requirements classification (must have/nice to have/deleted), priority and traceability information.

2. Overall Description

2.1. Product perspective

**Interfaces:** DocVisit portal is planned to operate online, enabling access from all major browsers. Interface would also be mobile and tablet friendly. As the system would not be used daily (ie. mainly for health sprints) then development would not include a special app.

**Users:** System users are in the following segments:

- a. Family Doctor
- b. Family Nurse
- c. Patient
  - i. Can also be “Representative”
- d. Represented person
- e. System Administrators

**Hardware:** System should run as an online environment. As there would be a login needed, then users might need to use ID-card reader in addition to PC (or use other identification method).
**Software:** All major web browsers applicable. System should follow the Responsive Web Design (RWD) criteria - an approach to web design that makes web pages render well on a variety of devices and window or screen sizes (Responsive Web Design – What It Is and How to Use IT, 2011).

**Hardware constraints:** System and its database need to be protected in accordance to applicable laws to handle sensitive personal data.

### 2.2. User characteristics

There are four main user groups of the DocVisit portal:

1) **Patient** – a person going to the family doctor appointment, meaning whoever living in Estonia and included to the family doctor list. Patient’s main activities in DocVisit: looking for suitable free slots for doctor’s appointment, cancelling/changing their appointment times, making prepayments for the visit (if not insured by the Health Insurance Fund, or needing service that is not covered by the Health Insurance Fund), looking for their appointments and medical history.

Patient can also be a **Representative** who manages his/her represented person(s) appointments and data.

2) **Represented person** – similar to the patient, but with different user access rights. Difference from the patient is that represented person has a representative who has user rights for his/her account. Represented person user rights are restricted allowing only viewing his/her medical data.

3) **Doctor** – Family Doctors working in specialised health care centres who need to see and maintain the information about their appointment calendars, see and add information regarding their patients treatments and visits.

4) **Nurse** – Family Nurses working in specialised health care centres with family doctors. Role is similar to Doctor’s role, but might have more restricted access rights. This will be discussed and decided in the later stages of the requirements specification process.

5) **System administrator** – smaller group of people, having access to the DocVisit are system administrators and developers, who need to handle and manage system errors, take care of development items and are responsible for further system maintenance.

To give an overview of most important relationships between main actors at the moment (without DocVisit system) we have chosen to analyse dependencies between Patient, Doctor and Nurse. In our opinion without new portal these 3 actors all play biggest roles in the process and must be included in the model.
Model is done by using i* modelling language and shows only strategic dependencies. Results can be seen on Graph 1 Model 1.

**Graph 1 Model 1:** Actor/stakeholder dependency model without DocVisit.

To illustrate how DocVisit will change relationships between actors and help to achieve goals we created another model where DocVisit portal is included. In our view DocVisit has two main counterparts – Patient and Medical Staff. Even though portal has more user groups (5 in total) other user groups have either smaller roles or difference is mainly in more restricted user rights. One of the major advantages of the portal is that both Doctor and Nurse will have similar access rights to the portal which means that they can both import, update and search for required data. Due to that clear dependency in relationships between Nurse and Doctor will be lost and these two roles will support each other based on ad hoc needs. Based on that following model shows relationships between three actors: Patient, DocVisit portal and Medical Staff. Medical Staff in this context includes both Doctor and Nurse roles, Patient includes Representative and Represented roles.

Model is done by using i* modelling language and shows only strategic dependencies. Results can be seen on Graph 2 Model 2.
Graph 2 Model 2: Actor/stakeholder dependency model including DocVisit.

2.3. Product functions

2.3.1 Major use cases

To illustrate further Medical Staff and Patients interaction with DocVisit portal (based on the Graph 2, Model 2) Use Case diagram was created (Graph 3).

Use Case diagram shows on a higher level how chosen actors interact with the system. Detailed level description of Use Cases (UC2, UC3, UC5, UC6, UC7) is provided in the following Use Case templates.

Following Use Cases are connected to Functional Requirements where system interacts either with Medical Staff (Doctor, Nurse) or Patient (including Representative or Represented person). Connection between Functional Requirement and Use Case can be found in Specific Requirements (Table 1) and Traceability Matrix (Table 2).
Graph 3 Use Case diagram

The most important use cases UC2, UC3, UC5, UC6, UC7 were selected and following Use Case templates filled for those:

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>UC 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case name</td>
<td>Add a new visit</td>
</tr>
<tr>
<td>Created By</td>
<td>Team 6</td>
</tr>
<tr>
<td>Last Updated By</td>
<td>Team 6</td>
</tr>
<tr>
<td>Date Created</td>
<td>12.11.2018</td>
</tr>
<tr>
<td>Date Last Updated</td>
<td>14.11.2018</td>
</tr>
<tr>
<td>Actors</td>
<td>Patient</td>
</tr>
<tr>
<td>Description</td>
<td>Patient uses system to add new visit to the Doctors office.</td>
</tr>
<tr>
<td>Trigger</td>
<td>Patient wants to visit a doctor.</td>
</tr>
</tbody>
</table>
| Precondition | • The patient is logged in to DocVisit.  
• System shows available time slots. |
| Post-condition | The system confirms the booking. |
| Normal flow: | 1. The patient selects „Book a visit“;  
2. System displays patient data; |
3. System displays new visit entry form with available time slots;
4. The patient selects suitable time;
5. Patient clicks „Add a visit“ button;
6. System controls whether the patient has HIF insurance.
7. System creates visit record and sends confirmation by e-mail to the user;
8. System changes selected time status to not available;
9. System saves booked time with patient’s details;
10. System sends an email reminder to the patient 3 days before the visit.

**Alternative flows**
If the patient cannot find the appropriate visit time in step 3:
4. a.1 Patient presses the button "Do not find the right time? “
4. a.2 The system displays the appropriate help text.

If patient doesn’t have HIF insurance in step 6:
7.a.1 System goes to UC7 step 3.

**Exceptions**
- Medical Staff has not updated appointments calendar →
  3.a.1 System displays message “No available times at the moment. Please call Doctors Office if needed.”
- User cannot add a new appointment (for himself/herself) if there’s an existing active one →
  5.a.1 System displays message “Adding new visit is not allowed when there is existing active booking.”
- System doesn’t send a confirmation by email
- System doesn’t send a reminder by email
- System doesn’t save patient details for the booked time →
  8.a.1 System displays error message “Data not saved”;
- System doesn’t change booked time status to not available

**Includes**
UC6; UC7

**Priority**
High
<table>
<thead>
<tr>
<th>Trigger</th>
<th>Patient wants to change an existing appointment.</th>
</tr>
</thead>
</table>
| Precondition | - The patient is logged in to DocVisit  
- The patient has an existing booking  
- The patient initiates the change for an existing appointment more than 3 days before appointment time slot. |
| Post-condition | System changes visit time. |
| Normal flow: | 1. System displays existing appointment details;  
2. The patient selects „Change visit“;  
3. System displays available time slots;  
4. The patient selects suitable new time;  
5. Patient clicks „Confirm new time“ button;  
6. System cancels the existing booking and makes it available again;  
7. System creates new visit record and sends confirmation by e-mail to the user;  
8. System changes selected new time status to not available;  
9. System saves booked time with patient details;  
10. System sends an email reminder to the patient 3 days before the visit. |
| Alternative flows | If there are less than 3 days till the appointment:  
2.a.1 System doesn’t show “Change visit” option  
2.a.2 System displays a message “Appointment changing is not possible anymore. If needed, please call Doctors Office by phone.”  
If the patient cannot find the appropriate visit time in step 3:  
4.a.1 Patient presses the button "Do not find the right time?“  
4.a.2 The system displays the appropriate help text. |
| Exceptions | - Medical Staff has not updated appointments calendar →  
3.a.1 System displays message “No available times at the moment. Please call Doctors Office if needed.”  
- System doesn’t save the changes →  
6.a.1 System displays error message “Data not saved” and further instructions.  
- System doesn’t send a confirmation by email →  
7.a.1 System creates error in the logs and sends it to system administrator.  
- System doesn’t change new booked time status to not available →  
8.a.1 System creates error in the logs and sends it to system administrator.  
- System doesn’t save patient details for the booked new time →  
9.a.1 System creates error in the logs and sends it to system administrator.  
- System doesn’t send a reminder by email → |
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>10.a.1 System creates error in the logs and sends it to system administrator.</strong></td>
<td></td>
</tr>
<tr>
<td>Includes</td>
<td>UC2</td>
</tr>
<tr>
<td>Priority</td>
<td>Medium</td>
</tr>
</tbody>
</table>

| **Use Case ID** | UC 5 |
| **Use Case name** | Manage patient record |
| **Created By** | Team 6 | **Last Updated By** | Team 6 |
| **Date Created** | 12.11.2018 | **Date Last Updated** | 15.11.2018 |
| **Actors** | Medical Staff (including Doctor and Nurse roles) |
| **Description** | Medical Staff uses system to manage patient record. |
| **Trigger** | Medical Staff wants to manage patient’s medical data. |
| **Precondition** | - The Medical Staff is logged in to DocVisit  
- System has patient’s records from previous visit(s)  
- Medical Staff sees only their patients’ data |
| **Post-condition** | System updates patient records. |
| **Normal flow:** | 1. The user selects manage patient record;  
2. System asks patient personal identification code (ID);  
3. User inserts patient ID;  
4. System displays patient records;  
5. System shows “Update” option;  
6. User selects “Update”;  
7. User enters new data and clicks “Save”;  
8. System updates patient record. |
| **Alternative flows** | If the user does not fill all required fields in step 7:  
8.a.1 System stops medical record updating action;  
8.a.2 System highlights required fields;  
8.a.3 User enters required data;  
8.a.4 User clicks “Save”.  
System proceeds with step 8 when user fills all required fields.  
If the user decides to search patient by first/last name:  
2.a.1 User selects search criteria as first/last name;  
2.a.2 User enters a patient first/last name;  
2.a.3 User clicks “Search”;  
System displays a list of patient records having similar first/last name. |
If patient personal identification code does not exist in the system in step 3:
4.a.1 System displays a message “ID incorrect”;
4.a.2 System re-asks for patient ID

**Exceptions**

- System doesn’t display patient record →
  4.a.1 System displays error message “Error - Patient data is not available.” System displays further instructions.
- System doesn’t allow updating patient record
  5.a.1 System displays error message “Error – Update option is not available.” System displays further instructions.

**Includes**

- 

**Priority**

- High

---

**Use Case ID**

UC 6

**Use Case name**

View medical history

**Created By**

Team 6

**Last Updated By**

Team 6

**Date Created**

12.11.2018

**Date Last Updated**

14.11.2018

**Actors**

Patient

**Description**

Patient uses system to view his/her medical data.

**Trigger**

Patient wants to view his/her medical history.

**Precondition**

- Patient is logged in to DocVisit.
- Medical Staff has created and updated patient’s medical record.

**Post-condition**

System displays medical data to the user.

**Normal flow:**

1. The user selects “View medical data”;
2. The user selects a time period;
3. System displays patient medical data in chronological order for the selected period.

**Alternative flows**

If there is no medical data for the selected time period:

3.a.1 System displays a message “No Medical History available for the selected period”;
3.a.2 Patient selects a new time period.

**Exceptions**

- System doesn’t display patient record →
<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>UC 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case name</td>
<td>Pay visit fees</td>
</tr>
<tr>
<td>Created By</td>
<td>Team 6</td>
</tr>
<tr>
<td>Date Created</td>
<td>12.11.2018</td>
</tr>
<tr>
<td>Actors</td>
<td>Patient</td>
</tr>
<tr>
<td>Description</td>
<td>Patient uses the system to prepay the visit fee, if needed.</td>
</tr>
<tr>
<td>Trigger</td>
<td>Patient needs to pay for the visit and prefers to do it online before the visit.</td>
</tr>
</tbody>
</table>
| Precondition | • The user is logged on to the system.  
• The user doesn’t have HIF insurance.  
• System displays the price for the visit.  
• System has online payment options available. |
| Post-condition | The user has paid for the visit. |
| Normal flow: | 1. The user selects the appointment.  
2. System controls if the patient has HIF insurance.  
3. The system displays the visit fee.  
4. The user selects online payment method.  
5. The system initiates the payment process.  
6. Payment reaches HIF.  
7. The system confirms the payment. |
| Alternative flows | If selected medical service is free of charge:  
3.a.1 System goes to UC 1 step 6.  
If the user doesn’t select an online payment  
5.a.1 System creates an unpaid invoice.  
5.a.2 Patient pays the invoice at the doctor office. |
| Exceptions | • Online payment doesn’t work → 4.a.1 System displays error message “Online payment not available” and provides further instructions. System goes to step 5.a.1.  
• Data exchange with HIF systems (via X-tee) doesn’t work, and insurance status cannot be checked → 2.a.1 System displays error message “Invalid query, please try again later”. |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>Low</td>
</tr>
<tr>
<td>Includes</td>
<td>UC2</td>
</tr>
</tbody>
</table>

2.3.2 UC2 refinement

As UC2 (Add a new visit) is one of the main and most important functions of DocVisit portal, it has been chosen for further refinement during which we will show in more details what classes are included in this UC, how these classes are connected and what is the sequence of the activities between these classes. To achieve the goal of more detailed refinement for UC2 following diagrams were created: class diagram (CD1), state diagrams (SD1, SD2, SD3) and sequence diagram (Seq. 1).

In the class diagram (Graph 7) there are four main classes: patient, medical staff, appointment and DocVisit portal. In addition to main classes patient has two sub-classes (representative and represented) and medical staff has also two sub-classes (nurse and doctor) which are based on the roles. CD1 shows relations between classes, sub-classes and main activities that are connecting these classes.
Based on the CD1 four main classes were defined for UC2. To show how objects these classes are changing their state three state diagrams were created: SD1 (Graph 8) – showing patient states, SD2 (Graph 9) – showing medical staff states and SD3 (Graph 10) – showing appointment states. State diagram for DocVisit portal was excluded in this step.

State diagrams are all showing any object (from each class) state changes during its lifecycle in the UC2 framework.
Graph 8 SD1 - State diagram of Patient during UC2
Finally to bring all classes, their relationships and order of activities together sequence diagram was created (Seq.1). Sequence diagram is showing UC2 main 4 classes and in which order activities between them take place. Sequence diagram of UC2 is shown on Graph 11.
2.4. Expectations

DocVisit portal should function as doctors or patient’s primary information resource. Portal should include patient medical history (treatment history, physical exam history, dispensed medications etc.), allergies, immunizations, diagnostic images and results, also appointment schedules. The system captures the patient health-related information needed for medical records, therefore paper medical records are no longer needed.

The system will increase efficiency, saves time and helps reducing the paperwork. Also, it helps to ease too time-consuming manual appointment reservation process.

2.4.1. System expectations

System should be user friendly and intuitive to use. Main users should not be required to obtain training before starting to use the portal. However, instructions and FAQ should be available and easily found at the portal.
DocVisit portal must be user friendly also for disabled persons, e.g. text should be visible for the visually impaired patients. The application layer must provide clear, informative and accurate error messages to the user.

### 2.4.2. Software expectations

System needs to be accessible 24h a day for all users and database should be secured.

DocVisit portal should organize and prioritize patient-related communication such as messages, diagnostic results, immunizations time etc. Portal should also support the management of communication.

System provides tools to support the work of the family doctors and nurses (tool for planning, organizing, monitoring etc.). The system should also capture data used for quality improvement, resource planning, performance management, and risk management.
# 3. Specific Requirements

Table 1. Specific Requirements

<table>
<thead>
<tr>
<th>Sub-goal ID</th>
<th>Req. ID</th>
<th>Requirement name</th>
<th>Use Case ID</th>
<th>Category: Mandatory (M)/ Nice to have (N)/Deleted</th>
<th>Priority H(high)/ M(medium)/L(low)</th>
<th>Traceability relationships</th>
<th>Towards the goals (ID)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td><strong>3.1 External Interface Requirements</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Req. 01</td>
<td>System should require user identification.</td>
<td>M</td>
<td>M</td>
<td>based_on</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Req. 02</td>
<td>System should support user identification by ID card.</td>
<td>M</td>
<td>M</td>
<td>satisfies</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Req. 03</td>
<td>System should support user identification by Mobile ID.</td>
<td>M</td>
<td>M</td>
<td>satisfies</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Req. 04</td>
<td>When user is logged in, the system should automatically connect patients to their doctors, so user sees only his/her applicable contacts.</td>
<td>N</td>
<td>L</td>
<td>Refines based_on</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Req. 05</td>
<td>Time booking option and observing one’s medical history should be separated in the system user interface.</td>
<td>N</td>
<td>L</td>
<td>based_on</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Req. 06</td>
<td>Representative should have user rights of the represented person.</td>
<td>M</td>
<td>M</td>
<td>based_on</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Req. 07</td>
<td>Represented persons can log into the system to view their medical history, but can’t take any actions in the system.</td>
<td>N</td>
<td>L</td>
<td>derived</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

<p>|             |        | <strong>3.2 System Requirements</strong>                                                      |             |                                                   |                                   |                          |                       |
| 6           | Req. 08| The system should be compliant with security standards applicable for handling sensitive personal and medical data. | M           | M                                                 | based_on                         | 6                        |
| 6           | Req. 08.1| The application should comply with a standard EVS 8:2008.                        | M           | M                                                 | Refines                          | 6                        |
| 6           | Req. 08.2| The system security class should be K2T2S2. (ISKE)                              | M           | M                                                 | Refines                          | 6                        |</p>
<table>
<thead>
<tr>
<th>Sub-goal ID</th>
<th>Req. ID</th>
<th>Requirement name</th>
<th>Use Case ID</th>
<th>Category: Mandatory (M)/ Nice to have (N)/ Deleted</th>
<th>Priority H(high)/ M(medium)/ L(low)</th>
<th>Traceability relationships</th>
<th>Towards the goals (ID)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Req. 09</td>
<td>The system must be able to handle at least 500 concurrent end-users.</td>
<td>N</td>
<td>L</td>
<td>Deleted</td>
<td>Refines 5</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Req. 10</td>
<td>Preparing the page to display on the server at the normal load should not exceed 0.5 seconds.</td>
<td>N</td>
<td>L</td>
<td>based_on 5</td>
<td>satisfies 1</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Req. 11</td>
<td>System should respond to a query during 3 seconds.</td>
<td>M</td>
<td>L</td>
<td>based_on 5</td>
<td>satisfies 1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Req. 12</td>
<td>The application must use UTF-8 encoding.</td>
<td>M</td>
<td>L</td>
<td>satisfies 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Req. 13</td>
<td>The system must have a separate administration module.</td>
<td>M</td>
<td>M</td>
<td>based_on 6</td>
<td>Refines 5</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>Req. 15</td>
<td>The system should make provision for different levels of access.</td>
<td>M</td>
<td>M</td>
<td>based_on 6</td>
<td>satisfies 5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Req. 16</td>
<td>The system should comprise security levels to guard against unauthorized changes of medical data.</td>
<td>M</td>
<td>M</td>
<td>based_on 6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Req. 16.1</td>
<td>The system security level should be medium (M). <em>(ISKE)</em></td>
<td>M</td>
<td>M</td>
<td>Refines 6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Req. 17</td>
<td>The system must log user activities that involve data modifications.</td>
<td>M</td>
<td>M</td>
<td>based_on 6</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 3.3 Functional Software Requirements

<table>
<thead>
<tr>
<th>1</th>
<th>Req. 18</th>
<th>Patient should be able to search for available appointment slots.</th>
<th>UC2</th>
<th>M</th>
<th>H</th>
<th>satisfies 1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Req. 19</td>
<td>Patient should be able to change his/her existing appointments.</td>
<td>UC3</td>
<td>N</td>
<td>H</td>
<td>satisfies 1</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>Req. 20</td>
<td>Patient should be able to cancel his/her appointment.</td>
<td>UC4</td>
<td>M</td>
<td>H</td>
<td>satisfies 1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Req. 21</td>
<td>Patient should be able to see details about his/her previous appointments.</td>
<td>UC3, UC6</td>
<td>M</td>
<td>L</td>
<td>based_on 2</td>
<td>2</td>
</tr>
</tbody>
</table>

23
<table>
<thead>
<tr>
<th>Sub-goal ID</th>
<th>Req. ID</th>
<th>Requirement name</th>
<th>Use Case ID</th>
<th>Category: Mandatory (M)/ Nice to have (N)/Deleted</th>
<th>Priority H(high)/ M(medium)/L(low)</th>
<th>Traceability relationships</th>
<th>Towards the goals (ID)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Req. 22</td>
<td>Doctor/Nurse should be able to change slots to available/not available.</td>
<td>UC1</td>
<td>M</td>
<td>H</td>
<td>based_on</td>
<td>satisfies 5</td>
</tr>
<tr>
<td>4</td>
<td>Req. 23</td>
<td>Doctor/nurse should be allowed to enter new medical data.</td>
<td>UC5</td>
<td>M</td>
<td>H</td>
<td>based_on</td>
<td>satisfies 5</td>
</tr>
<tr>
<td>4</td>
<td>Req. 24</td>
<td>Doctor/nurse should be able to trace a patient on his/her personal ID code.</td>
<td>UC5</td>
<td>M</td>
<td>H</td>
<td>based_on</td>
<td>satisfies 5</td>
</tr>
<tr>
<td>4</td>
<td>Req. 25</td>
<td>Doctor/nurse should be able to trace a patient on his/her surname.</td>
<td>UC5</td>
<td>M</td>
<td>H</td>
<td>based_on</td>
<td>satisfies 5</td>
</tr>
<tr>
<td>1</td>
<td>Req. 27</td>
<td>System should send automatic reminders to the patient of an upcoming appointment, when reminders are enabled by patient.</td>
<td>UC2, UC3</td>
<td>M</td>
<td>H</td>
<td>based_on</td>
<td>satisfies 5</td>
</tr>
<tr>
<td>3</td>
<td>Req. 28</td>
<td>System should allow patient to choose between different channels for reminders.</td>
<td>N</td>
<td>L</td>
<td></td>
<td>based_on</td>
<td>satisfies 5</td>
</tr>
<tr>
<td>1</td>
<td>Req. 29</td>
<td>System should enable making pre-payments for the doctor's visit.</td>
<td>UC7</td>
<td>N</td>
<td>H</td>
<td>based_on</td>
<td>satisfies 5</td>
</tr>
<tr>
<td>1</td>
<td>Req. 30</td>
<td>System should return the visit fee in case patient cancels the appointment.</td>
<td>UC4</td>
<td>N</td>
<td>H</td>
<td>based_on</td>
<td>satisfies 5</td>
</tr>
<tr>
<td>1</td>
<td>Req. 31</td>
<td>Available slots in the calendar should be updated in real time.</td>
<td>UC2, UC3</td>
<td>M</td>
<td>H</td>
<td>based_on</td>
<td>satisfies 5</td>
</tr>
<tr>
<td>6</td>
<td>Req. 32</td>
<td>System should enable different user roles based on 5 user classes.</td>
<td>M</td>
<td>M</td>
<td></td>
<td>based_on</td>
<td>satisfies 5</td>
</tr>
<tr>
<td>2</td>
<td>Req. 33</td>
<td>System should show patient’s history data in chronological order.</td>
<td>UC6</td>
<td>N</td>
<td>L</td>
<td>Refines</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>Req. 34</td>
<td>System should allow two ways for user to exit:</td>
<td>M</td>
<td>M</td>
<td></td>
<td>based_on</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>Req. 34,1</td>
<td>The user session is terminated when its session is exceeded by the timeout period;</td>
<td>M</td>
<td>M</td>
<td></td>
<td>Refines</td>
<td>6</td>
</tr>
<tr>
<td>Sub-goal ID</td>
<td>Req. ID</td>
<td>Requirement name</td>
<td>Use Case ID</td>
<td>Category: Mandatory (M)/ Nice to have (N)/Deleted</td>
<td>Priority H(high)/M(medium)/L(low)</td>
<td>Traceability relationships</td>
<td>Towards the goals (ID)</td>
</tr>
<tr>
<td>-------------</td>
<td>--------</td>
<td>------------------</td>
<td>-------------</td>
<td>--------------------------------------------------</td>
<td>----------------------------------</td>
<td>--------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>6</td>
<td>Req. 34.2</td>
<td>The user completes the session on his/her own initiative.</td>
<td>UC6</td>
<td>M M</td>
<td>Refines satisfies 6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Req. 35</td>
<td>The System should provide reports to the medical staff.</td>
<td>UC6</td>
<td>N H</td>
<td>based_on satisfies 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Req. 36</td>
<td>The system should allow printing the reports.</td>
<td>UC6</td>
<td>N H</td>
<td>based_on satisfies 5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 3.4 Non-Functional Software Requirements

<table>
<thead>
<tr>
<th>Req. 37</th>
<th>System should be usable for visually impaired people.</th>
<th>N L</th>
<th>Refines satisfies</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Req. 38</td>
<td>System should be available in 3 languages.</td>
<td>M L</td>
<td>based_on satisfies</td>
<td>3</td>
</tr>
<tr>
<td>Req. 38</td>
<td>System’s default language is Estonian.</td>
<td>M L</td>
<td>Refines satisfies</td>
<td>3</td>
</tr>
<tr>
<td>Req. 38</td>
<td>English and Russian language should be possible to choose.</td>
<td>M L</td>
<td>Refines satisfies</td>
<td>3</td>
</tr>
<tr>
<td>Req. 39</td>
<td>System should follow Responsive Web Design rules.</td>
<td>N L</td>
<td>based_on</td>
<td>5</td>
</tr>
<tr>
<td>Req. 40</td>
<td>System maintenance/scheduled downtime should be between 22.00-06.00.</td>
<td>M L</td>
<td>Refines satisfies</td>
<td>1</td>
</tr>
<tr>
<td>Req. 41</td>
<td>System maintenance/scheduled downtime should be announced a week in advance on the portal.</td>
<td>N L</td>
<td>Refines satisfies</td>
<td>1</td>
</tr>
<tr>
<td>Req. 42</td>
<td>System maintenance/scheduled downtime periods should not take place more often than twice a month.</td>
<td>N L</td>
<td>Refines satisfies</td>
<td>1</td>
</tr>
<tr>
<td>Req. 43</td>
<td>The system should be able to work when it’s components (database, application etc.) are located in different servers.</td>
<td>N L</td>
<td>based_on satisfies</td>
<td>5</td>
</tr>
<tr>
<td>Req. 46</td>
<td>System should run on all major browsers.</td>
<td>N L</td>
<td>based_on</td>
<td>5</td>
</tr>
<tr>
<td>Sub-goal ID</td>
<td>Req. ID</td>
<td>Requirement name</td>
<td>Use Case ID</td>
<td>Category: Mandatory (M)/ Nice to have (N)/Deleted</td>
</tr>
<tr>
<td>-------------</td>
<td>--------</td>
<td>----------------------------------------------------------------------------------</td>
<td>-------------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Deleted requirements</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Req. 14</td>
<td>Users without the necessary access rights should not be able to enter the</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>administrator module.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Req. 26</td>
<td>Doctor/nurse should be able to trace a patient on his/her address.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Req. 44</td>
<td>Created software must be compatible with existing hardware and software.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Req. 45</td>
<td>DocVisit is an online system; no data should be saved into any users’ local</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>computer.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Req. 47</td>
<td>System should be accessible where needed (remote access).</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4. Requirements Prioritization

4.1 Requirements hierarchy

Before requirements prioritization, an evaluation was performed where requirements were categorized into following groups: ‘must have’, nice to have’ and ‘delete’. As a result, 5 initial requirements were deleted, and they are not included in the following hierarchy. Requirements categorization can be seen in the Requirements Traceability table (Table 2).

DocVisit system’s primary goal is to enhance and improve family doctors’ and patients’ interaction by bringing time and cost savings.

System’s main goal can be achieved via the following sub-goals:

1. Saving patient’s time (Goal 1, ID1) 
2. Storing patient data in one place (Goal 2, ID2)
3. Personalizing patient's DocVisit usage (Goal 3, ID3)
4. Enabling time and cost saving by digitizing patient’s data (Goal 4, ID4)
5. User friendly system (Goal 5, ID5)
6. Confidential, secure DocVisit system (Goal 6, ID6)

The requirements were assembled in a hierarchical order based on 6 system sub-goals, and as a result the following requirements hierarchy (tree) was drawn (see Graph 4).
Enhancing and improving family doctors' and patients' interaction by bringing time and cost savings

<table>
<thead>
<tr>
<th>Saving patient's time (Goal/ID 1)</th>
<th>Storing patient data in one place (Goal/ID 2)</th>
<th>Personalizing patient's DocVisit usage (Goal/ID 3)</th>
<th>Enabling time and cost saving by digitizing patients data (Goal/ID 4)</th>
<th>User friendly system (Goal/ID 5)</th>
<th>Confidential secure DocVisit system (Goal/ID 6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Req. 18 Patient should be able to search for available appointment slots.</td>
<td>Req. 19 Patient should be able to change his/her existing appointments.</td>
<td>Req. 20 Time booking option and observing one’s medical history should be separated in the system user interface.</td>
<td>Req. 03 Time booking option and observing one’s medical history should be separated in the system user interface.</td>
<td>Req. 01 System should require user identification.</td>
<td>Req. 02 System should support user identification by ID card.</td>
</tr>
<tr>
<td>Req. 21 Patient should be able to see details about his/her previous appointments.</td>
<td>Req. 22 Patient should be able to change slots to available/not available.</td>
<td>Req. 04 When user is logged in, the system should automatically connect patients to their doctors, so user sees only his/her applicable contacts.</td>
<td>Req. 05 The system must be able to handle at least 500 concurrent end-users.</td>
<td>Req. 03 System should support user identification by Mobile ID.</td>
<td>Req. 06 Representative should have user rights of the represented person.</td>
</tr>
<tr>
<td>Req. 23 System should show patient’s history data in chronological order.</td>
<td>Req. 07 Represented persons can log into the system to view their medical history, but can’t take any actions in the system.</td>
<td>Req. 35 The System should provide reports to the medical staff.</td>
<td>Req. 09 The system must be able to print the reports.</td>
<td>Req. 08 The system should be compliant with security standards applicable for handling sensitive personal and medical data.</td>
<td>Req. 08.1 The application should comply with a standard EVS 8:2008.</td>
</tr>
<tr>
<td>Req. 24 Doctor/nurse should be able to trace a patient on his/her personal ID code.</td>
<td>Req. 36 The system should allow printing the reports.</td>
<td>Req. 06 Representative should have user rights of the represented person.</td>
<td>Req. 10 Preparing the page to display on the server at the normal load should not exceed 0.5 seconds.</td>
<td>Req. 08.2 The system security class should be K2T2S2. (ISKE)</td>
<td>Req. 13 The system must have a separate administration module.</td>
</tr>
<tr>
<td>Req. 25 Doctor/nurse should be allowed to enter new medical data.</td>
<td>Req. 11 System should respond to a query during 3 seconds.</td>
<td>Req. 07 The application must use UTF-8 encoding.</td>
<td>Req. 39 System should follow Responsive Web Design rules.</td>
<td>Req. 14 The system should make provision for different levels of access.</td>
<td>Req. 15 The system should log user activities that involve data modifications.</td>
</tr>
<tr>
<td>Req. 26 System should allow patient to choose between different channels for reminders.</td>
<td>Req. 27 System should allow patient to choose between different channels for reminders.</td>
<td>Req. 28 The application must use UTF-8 encoding.</td>
<td>Req. 40 System maintenance/scheduled downtime should be between 22.00-06.00.</td>
<td>Req. 32 System should enable different user roles based on 5 user classes.</td>
<td>Req. 43 The system should be able to work when it’s components (database, application etc.) are located in different servers.</td>
</tr>
<tr>
<td>Req. 28 System should allow patient to choose between different channels for reminders.</td>
<td>Req. 29 System should enable making pre-payments for the doctor’s visit.</td>
<td>Req. 30 System should return the visit fee in case patient cancels the appointment.</td>
<td>Req. 41 System maintenance/scheduled downtime should be announced a week in advance on the portal.</td>
<td>Req. 33 System should run on all major browsers.</td>
<td>Req. 46 System should allow two ways for user to exit</td>
</tr>
<tr>
<td>Req. 31 Available slots in the calendar should be updated in real time.</td>
<td>Req. 32 System should allow patient to choose between different channels for reminders.</td>
<td>Req. 33 System should show patient’s history data in chronological order.</td>
<td>Req. 42 System maintenance/scheduled downtime periods should not take place more than 2 times a week.</td>
<td>Req. 34 System should return the visit fee in case patient cancels the appointment.</td>
<td>Req. 34.1 The user session is terminated when its session is exceeded by the timeout period;</td>
</tr>
<tr>
<td>Req. 34 System should run on all major browsers.</td>
<td>Req. 35 System should show patient’s history data in chronological order.</td>
<td>Req. 36 The system should allow printing the reports.</td>
<td>Req. 43 The system should be able to work when it’s components (database, application etc.) are located in different servers.</td>
<td>Req. 34.2 The user completes the session on his/her own initiative.</td>
<td>Req. 34.3 The user completes the session on his/her own initiative.</td>
</tr>
</tbody>
</table>
4.2 Requirements prioritization

In order to prioritize the requirements, Analytic Hierarchy Process (AHP) calculations were performed based on Value and Cost. The calculations can be seen below.

### Value

**Step 1**

<table>
<thead>
<tr>
<th>Goal</th>
<th>Goal 1 (ID1)</th>
<th>Goal 2 (ID2)</th>
<th>Goal 3 (ID3)</th>
<th>Goal 4 (ID4)</th>
<th>Goal 5 (ID5)</th>
<th>Goal 6 (ID6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal 1 (ID1)</td>
<td>1.00</td>
<td>5.00</td>
<td>9.00</td>
<td>3.00</td>
<td>7.00</td>
<td>3.00</td>
</tr>
<tr>
<td>Goal 2 (ID2)</td>
<td>0.20</td>
<td>1.00</td>
<td>3.00</td>
<td>0.33</td>
<td>3.00</td>
<td>0.33</td>
</tr>
<tr>
<td>Goal 3 (ID3)</td>
<td>0.11</td>
<td>0.33</td>
<td>1.00</td>
<td>0.14</td>
<td>0.33</td>
<td>0.11</td>
</tr>
<tr>
<td>Goal 4 (ID4)</td>
<td>0.33</td>
<td>3.00</td>
<td>7.00</td>
<td>1.00</td>
<td>5.00</td>
<td>0.33</td>
</tr>
<tr>
<td>Goal 5 (ID5)</td>
<td>0.14</td>
<td>0.33</td>
<td>3.00</td>
<td>0.20</td>
<td>1.00</td>
<td>0.14</td>
</tr>
<tr>
<td>Goal 6 (ID6)</td>
<td>0.33</td>
<td>3.00</td>
<td>9.00</td>
<td>3.00</td>
<td>7.00</td>
<td>1.00</td>
</tr>
<tr>
<td><strong>Totals</strong></td>
<td><strong>2.11</strong></td>
<td><strong>12.66</strong></td>
<td><strong>32.00</strong></td>
<td><strong>7.67</strong></td>
<td><strong>23.33</strong></td>
<td><strong>4.91</strong></td>
</tr>
</tbody>
</table>

**Step 2**

<table>
<thead>
<tr>
<th>Goal</th>
<th>Goal 1 (ID1)</th>
<th>Goal 2 (ID2)</th>
<th>Goal 3 (ID3)</th>
<th>Goal 4 (ID4)</th>
<th>Goal 5 (ID5)</th>
<th>Goal 6 (ID6)</th>
<th><strong>Total</strong></th>
<th><strong>Value %</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal 1 (ID1)</td>
<td>0.47</td>
<td>0.39</td>
<td>0.28</td>
<td>0.39</td>
<td>0.30</td>
<td>0.61</td>
<td><strong>2.45</strong></td>
<td><strong>41%</strong></td>
</tr>
<tr>
<td>Goal 2 (ID2)</td>
<td>0.09</td>
<td>0.08</td>
<td>0.09</td>
<td>0.04</td>
<td>0.13</td>
<td>0.07</td>
<td><strong>0.51</strong></td>
<td><strong>8%</strong></td>
</tr>
<tr>
<td>Goal 3 (ID3)</td>
<td>0.05</td>
<td>0.03</td>
<td>0.03</td>
<td>0.02</td>
<td>0.01</td>
<td>0.02</td>
<td><strong>0.16</strong></td>
<td><strong>3%</strong></td>
</tr>
<tr>
<td>Goal 4 (ID4)</td>
<td>0.16</td>
<td>0.24</td>
<td>0.22</td>
<td>0.13</td>
<td>0.21</td>
<td>0.07</td>
<td><strong>1.02</strong></td>
<td><strong>17%</strong></td>
</tr>
<tr>
<td>Goal 5 (ID5)</td>
<td>0.07</td>
<td>0.03</td>
<td>0.09</td>
<td>0.03</td>
<td>0.04</td>
<td>0.03</td>
<td><strong>0.28</strong></td>
<td><strong>5%</strong></td>
</tr>
<tr>
<td>Goal 6 (ID6)</td>
<td>0.16</td>
<td>0.24</td>
<td>0.28</td>
<td>0.39</td>
<td>0.30</td>
<td>0.20</td>
<td><strong>1.57</strong></td>
<td><strong>26%</strong></td>
</tr>
</tbody>
</table>

### Cost

**Step 1**

<table>
<thead>
<tr>
<th>Goal</th>
<th>Goal 1 (ID1)</th>
<th>Goal 2 (ID2)</th>
<th>Goal 3 (ID3)</th>
<th>Goal 4 (ID4)</th>
<th>Goal 5 (ID5)</th>
<th>Goal 6 (ID6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal 1 (ID1)</td>
<td>1.00</td>
<td>0.33</td>
<td>5.00</td>
<td>3.00</td>
<td>5.00</td>
<td>0.20</td>
</tr>
<tr>
<td>Goal 2 (ID2)</td>
<td>3.00</td>
<td>1.00</td>
<td>9.00</td>
<td>3.00</td>
<td>7.00</td>
<td>0.33</td>
</tr>
<tr>
<td>Goal 3 (ID3)</td>
<td>0.20</td>
<td>0.11</td>
<td>1.00</td>
<td>0.20</td>
<td>3.00</td>
<td>0.11</td>
</tr>
<tr>
<td>Goal 4 (ID4)</td>
<td>0.33</td>
<td>0.20</td>
<td>5.00</td>
<td>1.00</td>
<td>3.00</td>
<td>0.14</td>
</tr>
<tr>
<td>Goal 5 (ID5)</td>
<td>0.20</td>
<td>0.14</td>
<td>3.00</td>
<td>0.33</td>
<td>1.00</td>
<td>0.11</td>
</tr>
<tr>
<td>Goal 6 (ID6)</td>
<td>5.00</td>
<td>3.00</td>
<td>9.00</td>
<td>7.00</td>
<td>9.00</td>
<td>1.00</td>
</tr>
<tr>
<td><strong>Totals</strong></td>
<td><strong>9.73</strong></td>
<td><strong>4.78</strong></td>
<td><strong>32.00</strong></td>
<td><strong>16.53</strong></td>
<td><strong>25.33</strong></td>
<td><strong>1.89</strong></td>
</tr>
</tbody>
</table>

**Step 2**

<table>
<thead>
<tr>
<th>Goal</th>
<th>Goal 1 (ID1)</th>
<th>Goal 2 (ID2)</th>
<th>Goal 3 (ID3)</th>
<th>Goal 4 (ID4)</th>
<th>Goal 5 (ID5)</th>
<th>Goal 6 (ID6)</th>
<th><strong>Total</strong></th>
<th><strong>Cost %</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal 1 (ID1)</td>
<td>0.1</td>
<td>0.07</td>
<td>0.16</td>
<td>0.18</td>
<td>0.2</td>
<td>0.11</td>
<td><strong>0.81</strong></td>
<td><strong>14%</strong></td>
</tr>
<tr>
<td>Goal 2 (ID2)</td>
<td>0.31</td>
<td>0.21</td>
<td>0.28</td>
<td>0.3</td>
<td>0.28</td>
<td>0.17</td>
<td><strong>1.55</strong></td>
<td><strong>26%</strong></td>
</tr>
<tr>
<td>Goal 3 (ID3)</td>
<td>0.02</td>
<td>0.02</td>
<td>0.03</td>
<td>0.01</td>
<td>0.01</td>
<td>0.06</td>
<td><strong>0.16</strong></td>
<td><strong>3%</strong></td>
</tr>
<tr>
<td>Goal 4 (ID4)</td>
<td>0.03</td>
<td>0.04</td>
<td>0.16</td>
<td>0.06</td>
<td>0.12</td>
<td>0.07</td>
<td><strong>0.49</strong></td>
<td><strong>8%</strong></td>
</tr>
<tr>
<td>Goal 5 (ID5)</td>
<td>0.02</td>
<td>0.03</td>
<td>0.09</td>
<td>0.02</td>
<td>0.04</td>
<td>0.06</td>
<td><strong>0.26</strong></td>
<td><strong>4%</strong></td>
</tr>
<tr>
<td>Goal 6 (ID6)</td>
<td>0.51</td>
<td>0.63</td>
<td>0.28</td>
<td>0.42</td>
<td>0.36</td>
<td>0.53</td>
<td><strong>2.73</strong></td>
<td><strong>46%</strong></td>
</tr>
</tbody>
</table>
From graph 5, it’s visible that for DocVisit system high-priority goals are Goal 1 and Goal 4. So as the first priority, most effort should be put to develop functionalities for goal 1 (Saving patients time) and 4 (Enabling time and cost saving by digitizing patients data). Goal 6 (Providing confidential and secure system) is with medium priority and goals 3 (Personalizing patient's DocVisit usage), 5 (User friendly system) and 2 (Storing patient data in one place) are low-priority.

4.3 High and medium priority goal modelling

Based on earlier concluded prioritization of goals we have selected high and medium priority sub-goals for creating goal model. Chosen cub-goals are: 1. Save patient time (ID1); 4. Enabling time and cost saving by digitizing patients data (ID4); 6. Confidential secure system (ID6).

Using KAOS modelling language we have refined chosen sub-goals into smaller steps creating new goal hierarchy until responsible agent for the lowest level goal is found. DocVisit KAOS model is given on Graph 6 on a generalised level. Lower levels are given on Graphs 6.1, 6.2 and 6.3.

Requirements list (Table 1) has currently connection between requirement and sub-goal (goal ID’s 1-6). It was decided not to update connections between requirements and lower level goals as they are
easily traceable to sub-goals and adding new level of connection will unnecessarily complicate requirements template.

**Graph 6** Goal and Agent (responsibility) KAOS model on generalised level

**Graph 6.1** DocVisit KAOS model in detailed level for ID1: Saving patient’s time
Graph 6.2 DocVisit KAOS model in detailed level for ID4: Time and cost saving by digitalizing patient's data

Graph 6.3 DocVisit KAOS model in detailed level for ID6: Providing confident, secure system
5. Requirements Traceability

5.1 Requirements traceability model

To provide requirements traceability, sub-goals, use case scenarios, requirements, class diagrams, state diagrams and sequence diagrams were selected as traceability artefacts and together with traceability type indicating the traceability relationship combined into traceability model (graph 7).

*Graph 7 Traceability model*

Traceability types (based on, derived, refines, satisfies, precondition) are further described under following sub-sections together with traceability matrixes.
5.2 Requirements traceability towards goals

All ‘must have’ and ‘nice to have’ requirements (source artefacts) were evaluated against sub-goals (target artefacts) and traceability relationships documented in the traceability matrix in Table 2. Clarification on the traceability types used:

- Based on: requirement has influenced the definition of the Goal
- Derived: requirement was derived based on (a set of) other requirement(s)
- Refines: requirement defines the Goal in more detail
- Satisfies: realising the requirement supports achieving the Goal
Table 2. Requirements and goals traceability

<table>
<thead>
<tr>
<th>Goal/ID1</th>
<th>Goal/ID2</th>
<th>Goal/ID3</th>
<th>Goal/ID4</th>
<th>Goal/ID5</th>
<th>Goal/ID6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save patient time</td>
<td>Store patient data in one place</td>
<td>Personalize patient experience</td>
<td>Enabling time and cost saving by digitizing patients data</td>
<td>User friendly system</td>
<td>Confidential secure system</td>
</tr>
</tbody>
</table>

3.1 External Interface Requirements

Req. 01. The system should support user identification.

Req. 02. The system should support user identification by ID card.

Req. 03. The system should support user identification by Mobile ID.

Req. 04. When user is logged in, the system should automatically connect patients to their doctors, so user sees only his/her applicable contacts.

Req. 05. Time booking option and observing one’s medical history should be separated in the system user interface.

Req. 06. Representative should have user rights of the represented person.

Req. 07. Represented persons can log into the system to view their medical history, but can’t take any actions in the system.

3.2 System Requirements

Req. 08. The system should be compliant with security standards applicable for handling sensitive personal and medical data.

Req. 08.1. The application should comply with a standard EVS 8:2008.

Req. 08.2. The system security class should be K2T2S2. (ISKE)

Req. 09. The system must be able to handle at least 500 concurrent end-users.

Req. 10. Preparing the page to display on the server at the normal load should not exceed 0.5 seconds.

Req. 11. System should respond to a query during 3 seconds.

Req. 12. The application must use UTF-8 encoding.

Req. 13. The system must have a separate administration module.

Req. 14. The system should make provision for different levels of access.

Req. 15. The system security level should be medium (M) (ISKE).

Req. 16. The system should provide reports to the medical staff.

Req. 17. The system must log user activities that involve data modifications.

3.3 Functional Software Requirements

Req. 18. Patient should be able to search for available appointment slots.

Req. 19. Patient should be able to change his/her existing appointments.

Req. 20. Patient should be able to cancel his/her appointment.

Req. 21. Patient should be able to see details about his/her previous appointments.

Req. 22. Doctor/Nurse should be able to change slots to available/not available.

Req. 23. Doctor/nurse should be allowed to enter new medical data.

Req. 24. Doctor/nurse should be able to trace a patient on his/her personal ID code.

Req. 25. Doctor/nurse should be able to trace a patient on his/her surname.

Req. 26. System should send automatic reminders to the patient of an upcoming appointment, when reminders are enabled by patient.

Req. 27. System should allow patient to choose between different channels for reminders.

Req. 28. System should allow patient to choose between different channels for reminders.

Req. 29. System should enable making pre-payments for the doctor’s visit.

Req. 30. System should return the visit fee in case patient cancels the appointment.

Req. 31. Available slots in the calendar should be updated in real time.

Req. 32. System should enable different user roles based on 5 user classes.

Req. 33. System should show patient’s history data in chronological order.

Req. 34. System should allow printing the reports.

Req. 35. System should allow two ways for user to exit.

Req. 36. The user session is terminated when its session is exceeded by the timeout period.

Req. 37. The user completes the session on his/her own initiative.

Req. 38. The System should provide reports to the medical staff.

Req. 39. The system should support Responsive Web Design rules.

Req. 40. System maintenance/scheduled downtime should be between 22.00-06.00.

Req. 41. System maintenance/scheduled downtime should be announced a week in advance on the portal.

Req. 42. System maintenance/scheduled downtime periods should not take place more often than twice a month.

Req. 43. The system should be able to work when its components (database, application etc.) are located in different servers.

Req. 44. System should run on all major browsers.
5.3 Functional requirements traceability towards Use Cases

Functional requirements (source artefacts) and Use Case scenarios (target artefacts) traceability relationships are documented in traceability matrix in Table 3. Clarification on the traceability types used:

- Based on: requirement has influenced the definition of Use Case
- Satisfies: realising the requirement supports Use Case
- Refines: requirement defines the Use Case in more detail
- Precondition: realising the requirement is a precondition for Use Case

Table 3. Functional requirements and Use Case traceability

<table>
<thead>
<tr>
<th>Req. No.</th>
<th>Description</th>
<th>UC1: Provide available slots</th>
<th>UC2: Add new visit</th>
<th>UC3: Change existing appointment</th>
<th>UC4: Cancel existing appointment</th>
<th>UC5: Manage patient records</th>
<th>UC6: View medical history</th>
<th>UC7: Pay visit fee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Req. 18</td>
<td>Patient should be able to search for available appointment slots.</td>
<td>satisfies</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 19</td>
<td>Patient should be able to change his/her existing appointments.</td>
<td>satisfies</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 20</td>
<td>Patient should be able to cancel his/her appointment.</td>
<td>based on</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 21</td>
<td>Patient should be able to see details about his/her previous appointments.</td>
<td>satisfies</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 22</td>
<td>Doctor/Nurse should be able to change slots to available/not available.</td>
<td>based on precondition</td>
<td></td>
<td>satisfies</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 23</td>
<td>Doctor/nurse should be allowed to enter new medical data.</td>
<td>based on</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 24</td>
<td>Doctor/nurse should be able to trace a patient on his/her personal ID code.</td>
<td>satisfies</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 25</td>
<td>Doctor/nurse should be able to trace a patient on his/her surname.</td>
<td>satisfies</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 27</td>
<td>System should send automatic reminders to the patient of an upcoming appointment, when reminders are enabled by patient.</td>
<td>satisfies</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 29</td>
<td>System should enable making pre-payments for the doctor’s visit.</td>
<td>based on</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 30</td>
<td>System should return the visit fee in case patient cancels the appointment.</td>
<td>satisfies</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 31</td>
<td>Available slots in the calendar should be updated in real time.</td>
<td>satisfies</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 33</td>
<td>System should show patient’s history data in chronological order.</td>
<td>refines</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 35</td>
<td>The System should provide reports to the medical staff.</td>
<td>refines</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Req. 36</td>
<td>The system should allow printing the reports.</td>
<td>refines</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Revision list


First revision done by 15.10.2018. Revision included:

- Overview added (part 1.4).
- User characteristics (part 2.3) updated with 1 additional user group.
- Specific requirements (part 3) updated.

Second revision done by 29.10.2018. Revision included:

- Adding new part of the document for Requirements prioritization (part 4).
- Adding new part of the document for Requirements traceability (part 5).
  - Updated requirements list is presented in Table 2 (part 5.2) with updated category, priority and traceability information.

Third revision done by 5.11.2018. Revision included:

- Moving part 5.2 (Table 2) to part 3 to replace original list of requirements without additional information.
- Adding new part of the document for Actor modelling (part 6).
- Adding new part of the document for Goal modelling (part 7).

Fourth revision done by 12.11.2018. Revision included:

- Moving previously separate part (6) of actors relationship models to part 2.3
- Moving previously separate part (7) on goals modelling to part 4.3
- Updating overview (part 1.4)
- Updating models, matrix and text with actual ID’s (where were missed before)

Fifth revision done by 19.11.2018. Revision included:

- Updating Traceability matrix by adding Use Case IDs to Functional Software Requirements
- Updating Specific Requirements (Table 1) with Use Case ID’s
- Updating part 2.3 with Use Case Diagram and Use Case templates.

Sixth revision done by 26.11.2018. Revision included:
- Updating Use Case diagram syntax
- Updating part 5 (5.1-5.3) by adding a graph of traceability model and creating separate traceability matrixes for Goals and Use Case scenarios

Seventh revision done by 2.12.2018. Revision included:
- Adding part 2.3.1 showing UC2 classes, class objects states and sequence diagram
- Updating traceability model

Eighth revision done by 10.12.2018. Revision included:
- Updating part 2.3.2 – names of the diagrams, diagrams
- Updating traceability model in part 5.1
Reference list


  https://www.smashingmagazine.com/2011/01/guidelines-for-responsive-web-design/

  https://www.riigiteataja.ee/akt/13125331?leiaKehtiv


Used diagram resources

- Actor/Stakeholder dependency models were created using iWork and artboard 2
- Traceability model and sequence diagram were created using www.draw.io
- Use Case diagram was created using Lucidchart
- Goal and Agent (responsibility) KAOS models were created using Objectiver (Trial Version) http://www.objectiver.com