Deliverable No. 15.4

Second evaluation workshops round

Grant Agreement No.:	270089
Deliverable No.:	D15.4
Deliverable Name:	Second evaluation workshops round
Contractual Submission Date:	31/07/2013
Actual Submission Date:	12/09/2013

Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

COVER AND CONTROL PAGE OF DOCUMENT	
Project Acronym:	*p-medicine*
Project Full Name:	From data sharing and integration via VPH models to personalized medicine
Deliverable No.:	D 15.4
Document name:	Second evaluation workshops round
Nature (R, P, D, O)[1]	R
Dissemination Level (PU, PP, RE, CO)[2]	PU
Version:	3.0
Actual Submission Date:	12/09/2013
Editor: Institution: E-Mail:	Marie-Luise Christ-Neumann Fraunhofer IAIS marie-luise.christ-neumann@iais.fraunhofer.de

ABSTRACT:

Usability is a continue process in the development of the various tools in the p-medicine project. Therefore it is essentially necessary to test the tools as soon as they are available and at different development stages. These usability tests show the importance when external users who are not acquainted with the software tools take part in this process.

This second evaluation workshop presents the fruitful results of the usability tests with internal and external users partly concerning the second prototypes of the developed tools. It suggests to the developer how to improve the software and recognize the user requirements in order to achieve qualitative ergonomic and usable tools for all user groups.

KEYWORD LIST: Evaluation, use scenarios of end-users

The author is solely responsible for its content, it does not represent the opinion of the European Community and the Community is not responsible for any use that might be made of data appearing therein.

---

[1] **R**=Report, **P**=Prototype, **D**=Demonstrator, **O**=Other

[2] **PU**=Public, **PP**=Restricted to other programme participants (including the Commission Services), **RE**=Restricted to a group specified by the consortium (including the Commission Services), **CO**=Confidential, only for members of the consortium (including the Commission Services)

MODIFICATION CONTROL			
Version	Date	Status	Author
1.0	30/05/2013	Draft	Marie-Luise Christ-Neumann
1.1	22/07/2013	Draft	Marie-Luise Christ-Neumann
1.2	24/07/2013	Draft	Marie-Luise Christ-Neumann
1.3	29/07/2013	Draft	Simona Rossi
1.4	30/07/2013	Prefinal	Marie-Luise Christ-Neumann
2.0	31/07/2013	Prefinal	Marie-Luise Christ-Neumann
2.1	11/09/2013	Prefinal	Fatima Schera
2.2	11/09/2013	Prefinal	Simona Rossi
3.0	12/09/2013	Final	Marie-Luise Christ-Neumann

List of contributors

– Simona Rossi, SIB

– Marie-Luise Christ-Neumann, FhG IAIS

– Marian Taylor, UOXF

– Alberto Anguita, UPM

– Alessandra Gorini, ecancer

– Ketti Mazzocco, ecancer

– Holger Stenzhorn, USAAR

– Norbert Graf, USAAR

– Fatima Schera, FhG IBMT

– Gabriele Weiler, FhG IBMT

– Benjamin Jefferys, UCL

– Stefan Rüping, FhG IAIS

– Wouter Dhaeze, Custodix

– Stelios Sfakianakis, FORTH

Contents

1 Executive Summary

In addition to the evaluation and validation activities in Task 15.2, described in D15.2 this task concentrates on usability tests with internal and external prospective end-users of the first and second prototype of the clinical tool ObTiMA and the Ontology Annotator (OA). A first evaluation activity of the ALGA Questionnaire had been arranged and is also described in this document.

Having the requirements of the users in mind which are described in detail in context scenarios (D2.2) of the various user groups, the p-medicine platform is planned and several tools are developed to support the heterogeneous end-users groups in their daily work: clinicians, data managers, bioinformaticians, biologists, data miners and patients.

Usability plays an essential role, both the platform and the tool interfaces have to be conforming to ISO 9241, especially with part 11 and part 110. To reach this aim an iterative process of usability tests have to be taken into consideration during the whole term of the project p-medicine (Figure.1).
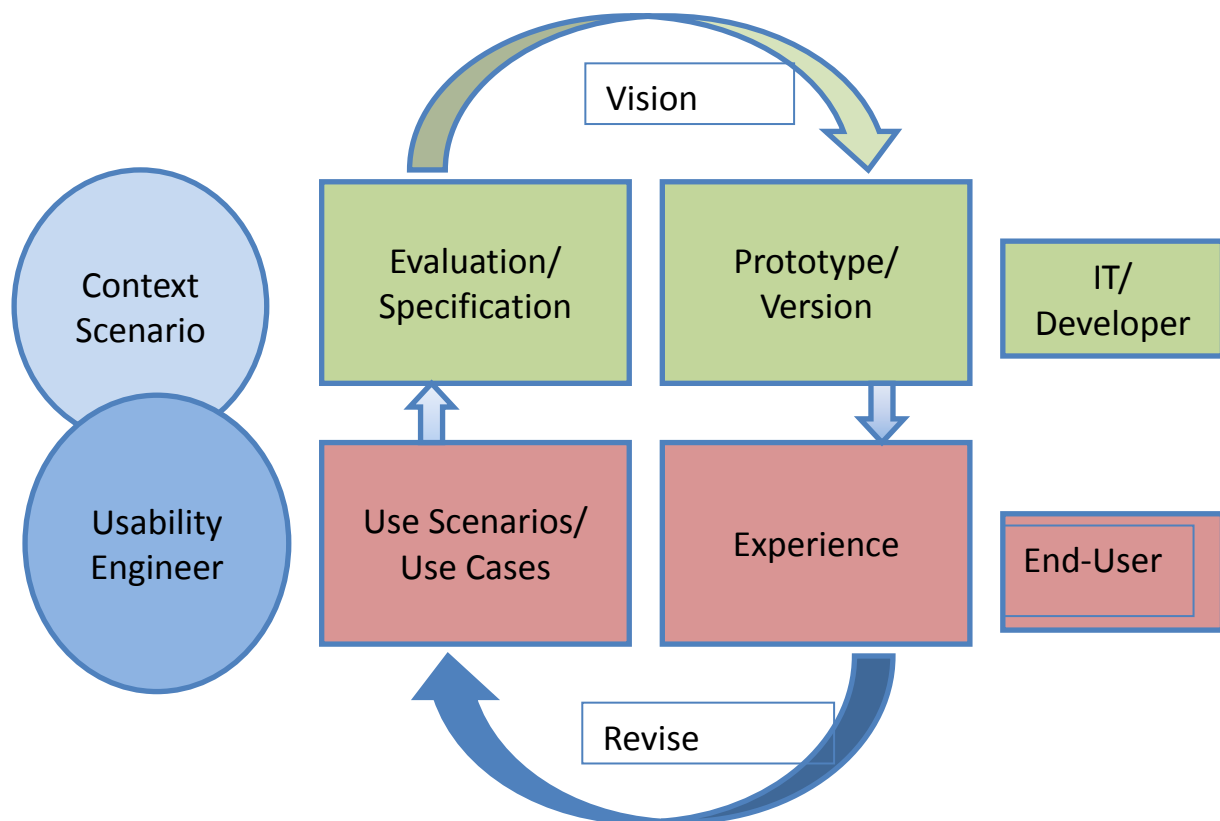
Figure 1: Development Loop of the iterative usability process

2 Introduction

The usability process in p-medicine takes the end-users, clinicians, data managers, bioinformaticians, biologists, data miners and patients into its focus for testing the developed prototypes of the tools. It is a continuous process which takes place during each development stage.

In order to guarantee ergonomic and qualitative usable software it is essentially necessary to start with first prototyping as early as possible. This prototyping process concerns all tools, modules and services in p-medicine. It gives the end-users the opportunity to get a first view and impression of the tool and it shows the developers what they have to improve or which requirements or needs have changed or have not been recognized to date. With the whole context of use and all relevant criteria describing how the user conducts his/her daily work task, described in context scenarios for each user group in D2.2 the designers developed first prototypes of the various tools. The aim of these prototypes is not to present the full functionality of the tool but rather show the user a first vision of the application. Additionally it shows the usability engineer if the comprehension of the users' needs were the same to the developers.

The patients who would like to interact via the portal will have a set of the corresponding tools that will allow *in primis* a better dialogue between them and the clinicians and therefore a better and concerted decision support. Therefore it is extremely significant to provide them an intuitive and comprehensible interface to share all needed information in an efficient way.

The ALGA-C questionnaire is a web-based tool targeted at p-medicine end-users that have the role of patient. Essentially, the ALGA-C questionnaire is used to collect psycho-cognitive information about patients.

The combined clinical and psycho-cognitive information becomes patient's and doctor's common knowledge, around which they can build together long term efficient decision-making plans.

A first evaluation of the ALGA Questionnaire will be described in chapter 5.
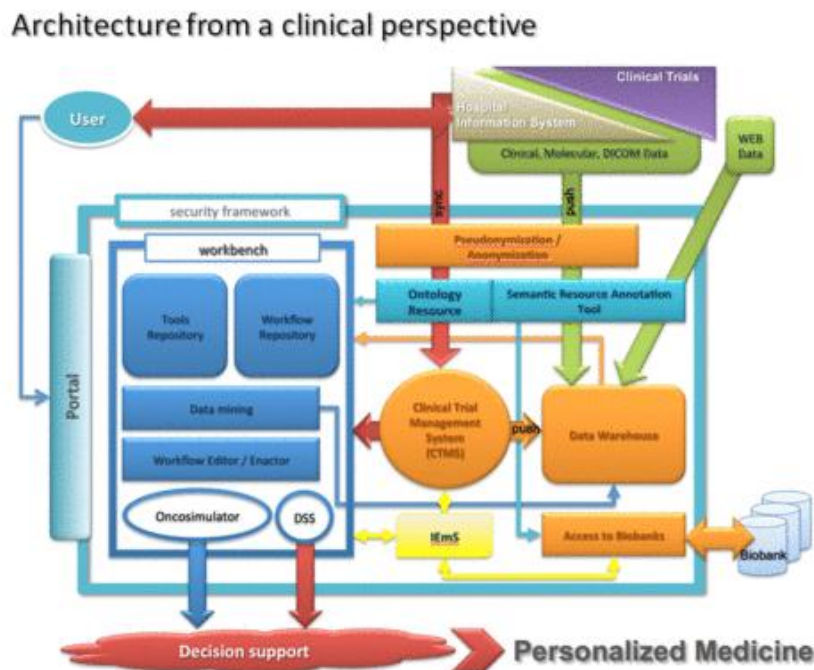
Figure 2: p-medicine infrastructure

Several usability tests were based on the ontology based clinical trial management application ObTiMA (chapter 3) and the Ontology Annotator (chapter 4). ObTiMA is the main clinical tool in p-medicine where various user groups are working. Figure 3 shows the starting page with a short introduction of the application ObTiMA.
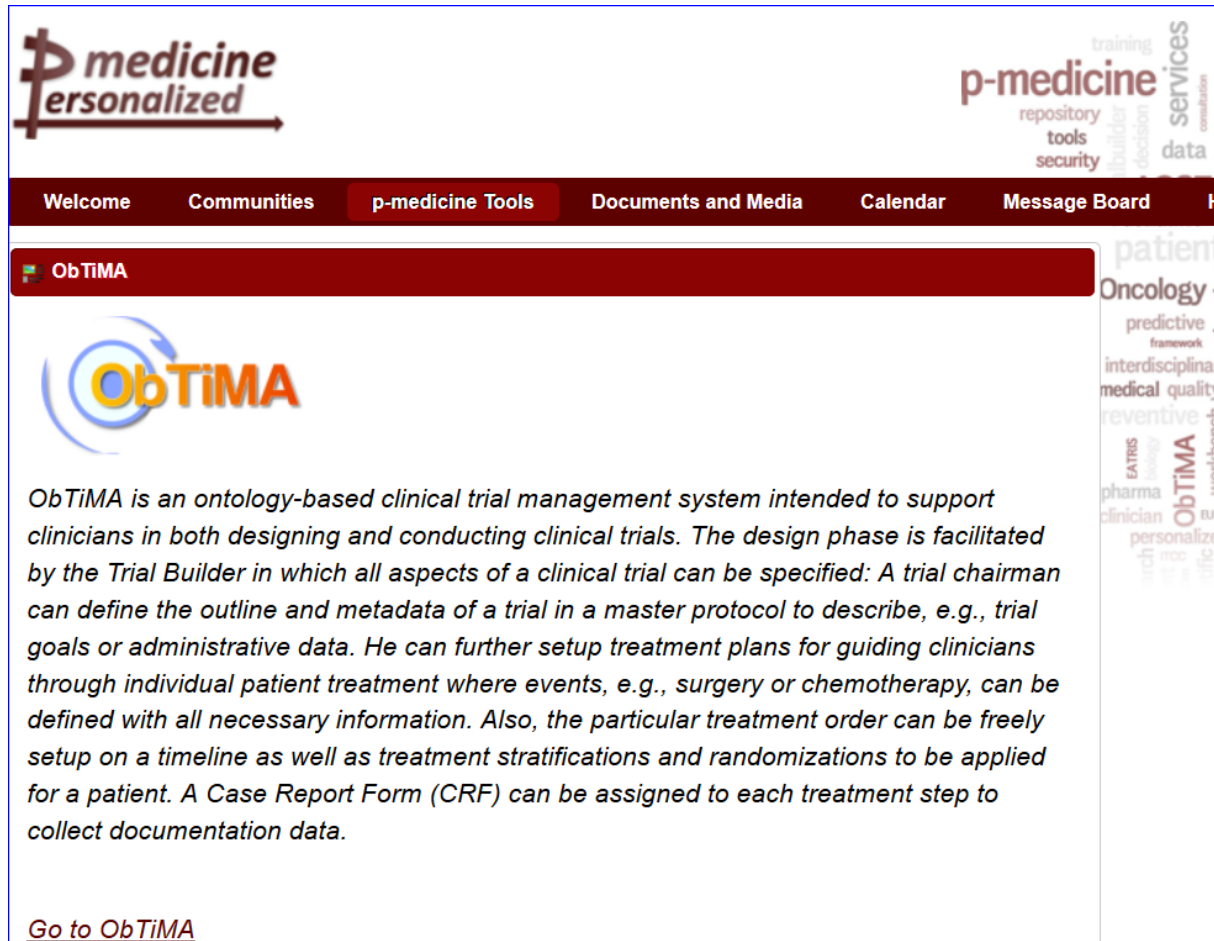
Figure 3: Starting page of ObTiMA

The first prototype of the Ontology Annotator (OA) has been tested by p-medicine members in a very early project phase with small functionality. It presents the user a first view of the interface. The detected deficiencies and problems the user allowed the developers to improve the software for the next prototype (cf. chapter 4).

3   Second usability tests of the clinical tool ObTiMA

ObTiMA is an ontology-based clinical trial management application, described in D2.2. The first prototype of the tool has been tested with prospective external users, a clinician and a study nurse, described in detail in D15.2. It shouldn't have the complete functionality but for the users it should demonstrate the well-understood requirements and for the developers to show the weaknesses and problems the user has to conduct her task efficiently. The detected deficiencies and enhanced requirements should be improved and considered in the second prototype that has been tested by prospective internal and external users again. Software design means to have a common understanding about the functionality the user needs to conduct the task efficiently.

At the end of year 2012 a data manager, member of p-medicine, tested the first prototype of the application tool ObTiMA again with the same task the external users had before. The task, entering patient data into a running trial is shown in Appendix A.

This test was not participatory observed by the usability engineer. With a detailed description the user was able to handle such a test herself. She got a description about the equipment she has to use and the task to conduct with the support of ObTiMA. A tool for recording her session is the free available tool CamStudio (http://camstudio.org). It serves as input for the evaluation process performed by the usability engineer. The result of this evaluation is important for the developers to improve the software if the user has problems to conduct the task in an efficient and satisfied way with the tool. A microphone is also necessary for the "thinking aloud" method that provides the usability engineer with information about the user's actions, his/her feelings, all emotions about frustration of the system behaviour or misunderstandings. All of these information result in form of use scenarios.

Two resulting use scenarios will follow. The corresponding screenshots are shown in Appendix B. A use scenario is more than a use case. It describes in four columns the task, the action of the user with comments of the user, the reaction of the application and in the fourth column the observed problems with recommendations of the usability engineer. In column three beside the action of the application "critical incidents" are described. A "critical incident" can be a problem, e.g. a misunderstanding of an error message or an unexpected reaction of the system or even other. The focus is on the ISO norm 9241, especially on the dialogue principles and the presentation of information. A complete description and the whole process of usability testing are described in D2.1.

The data manager conducted the task two times because in the first test she didn't achieve her supposed aim. She ran into a loop and didn't know how to get out of it. Therefore she conducted the test once more which showed again the same problem. Both use scenarios of the first prototype of ObTiMA are shown in the following:

First usability test: first prototype

First prototype of ObTiMA, tested by a data manager, member of p-medicine. The appropriate screen shots (Test ObTiMA M1) are shown in Appendix B.

Use Scenario – ObTiMA – M1 Screenshots in Appendix B, ObTiMA-M1				
Task decomposed into subtasks	**Step**	**User action and comment**	**Reaction of the dialogue system (ObTiMA)**	**Usage requirement, observed problems**
Key task: Entering data of a fictive patient.				**Usage requirement:** To enter patient data into Case Report Forms (CRFs) under an available trial efficiently. The developers should consider the ISO 9241 – 110: Dialogue principles which should be taken into consideration in the sense to conduct the user's task in an efficient, effective and satisfied way (ISO 9241 – 11).

Subtask 1:			Initial situation:	With the support of the ontology based clinical trial management application ObTiMA the user has the possibility to conduct her task to enrol patient data.
Login			The welcome page of ObTiMA is shown.	
	1.	User M logged in to ObTiMA under her account.	User M is logged in successfully.	
			The main menu with the tabs "Trials" "Patients" "Tools" and "Administration" are presented. (Figure B1M1)	
			A speech bubble is shown with the message: "Please select a task from the main menu!"	The user didn't know under which tab she has the possibility to enter patient data. The user needed more information in this situation.
Subtask 2:			A menu is opened with three items in different languages. (Figure B2M1)	Consistency of the used language. Has to be improved by the developers
To enrol patient data.	2.	She noticed the speech bubble text and clicked on the entry "Trials".		➔ Presentation of information (ISO 9241-12)
				The user can only enrol patient data into an existing trial. A recommendation for better comprehensibility of the name of the menu. It should have the name "Manage Patients under Trial". This shows the user that patient data can only be entered under an existing trial.
			A menu is opened with one entry "Manage Patients". (Figure B3M1)	
	3.	She clicked on the tab "Patients".		The breadcrumb in Figure B4M1 has a better name (Manage Trial Patients) as the menu item (Manage Patients). It shows the user only to enrol patient data under an existing trial.
			A new window is opened with one trial named "Nephroblastoma Therapy optimisation SIOP2001/GPOH. (Figure B4M1)	
	4.	She clicked on the entry "Manage Patients".		

Subtask 3: Looking for existing patients and for existing CRFs	5.	She selected the trial with the acronym SIOP2001/GPOH. *Comment:* "It shows the patients associated to the trial SIOP2001/GPOH." "I got the patient number and identification number but I had no pseudonym of the patient. So, I don't know if the pseudonym belongs to my patient I created some days ago." "I guess this is the same one."	A new window shows one entered patient with the pseudonym and birthdate. Additionally two buttons "Create Patient" and "Search for Patient" are presented. (Figure B5M1)	With the two buttons "Create Patient" and "Search for Patient" the user has the possibility to create a new patient or to search for a pseudonym. The user was wondering if this pseudonym is the patient she created some days before. She identified it as her patient she entered some days ago.
	6.	She clicked on the item "Birthdate".	Nothing happened!	The user has the only possibility to click on the pseudonym to see the content. To make the whole line mouse sensitive would be more intuitive for the user.
	7.	She clicked on the pseudonym. *Comment:* "It is my fictive patient 'Peter Maier-	The CRF for the patient with the entered data occurred. (Figure B6M1)	

		Landroth' and the organisation 'University Oxford'."		
	8.	User M scrolls in the list of the available languages and selected the language German. Then she clicked on the button "Save".	The name in the language field was changed to German. A short refresh by the system is shown but no other response is given. **Critical incident:** The user expected a response of the system. The short refresh gave the user not the confirmation of a save action.	The user entered the fictive patient some days ago with the language "Esperanto", she remembered when the recording system crashed. Now she changed the language into "German" successfully. The save action should be better visible by the user in form of a response action, e.g. "successfully saved". ➔ Controllability, Suitability for the task (ISO 9241 – 110)
	9.	The user wanted to look for the CRFs of the patient. She clicked on the button "CRF"	A new window occurred with the button "Add CRF" and the hint "Please select a CRF to input data for this patient". (Figure B7M1) **Critical incident:** There is no CRF to select.	There was no CRF that could be selected. For the user not intuitively clear to click on the button "Add CRFs" to see a CRF.
	10. 11.	She clicked on the button "Add CRFs". She marked the BaselineDataSIOP CRF by a check mark and clicked on it.	The new window shows the "BaselineDataSIOP CRF (Figure B8M1). A preview of the Demographics section of the CRF is shown on the new window. (Figure B9M1)	The user wanted to enter new data but she got only a preview. That was not what she expected. ➔ Conformity with user expectation.(ISO 9241-110)
	12.	She browsed through the sections.	A preview of section "Operation Details", the second section of the BaselineDataSIOP is shown. (Figure B10M1)	

	13.	She went back to the CRF via the breadcrumbs	The same window occurred as before (Figure B8M1).	The user was looking around to enter data respectively to assign the CRF to the patient.
	14.	She clicked on the button "Assign selected CRF to the patient" (Fig.6)	The window with the three buttons "Patient Data", "CRFs" and "Treatment Plan" occurred with the hint "Please select a CRF to input data for this patient".	There was no CRF that could be selected. The user was in a loop where she has no idea how to enter new data. **Bug of the system!** Step 9 shows the same situation.
	15:	She clicked on the button "Add CRFs". User M stopped the session.	The same window as in Figure B8M1 is shown.	The user has no idea to move out of this loop. Unintentionally the user saved the CRF "BaselineDataSIOP" two times. She stopped the session unsatisfied.

Second usability test: first prototype

First prototype of ObTiMA tested a second time by the data manager. She didn't achieve her aim efficiently in her first test.

Use Scenario – ObTiMA – M2				
Screenshots in Appendix B, ObTiMA-M2				
Task decomposed into subtasks	**Step**	**User action and comment**	**Reaction of the dialogue system (ObTiMA)**	**Usage requirement, observed problems**
Key task: Entering data of a fictive patient.				**Usage requirement:** To conduct the task with the support of the ontology based clinical trial management application ObTiMA. The user enrolled patient data into Case Report Forms (CRFs) under an available trial. The developers should consider the ISO 9241 – 110: Dialogue principles that should be taken into consideration in the sense to conduct the user's task in an efficient, effective and satisfied way (ISO 9241 – 11).

Subtask 1: Login **Subtask 2:** Looking for a fictive patient she entered a day before	1.	User M logged in to ObTiMA under her account.	**Initial situation:** The welcome page of ObTiMA is shown. User M is logged in successfully. The main menu with the tabs "Trials" "Patients" "Tools" and "Administration" are presented. (Figure B1M2)	The test takes place outside of the p-medicine portal. ObTiMA's website has been called via the user's browser. The user knows from the first test a day before to click on the tab "Patients".
	2.	She clicked on the tab "Patients".	A menu is opened with the entry "Manage Patients".	
	3.	She clicked on the button "Manage Patients".	On the new opened window two trials she created in the past are shown. (Figure B2M2)	
	4.	She selected the trial with the acronym SIOP2001/GPOH.	A new window shows one entered patient with the pseudonym and birthdate. Two buttons "Create Patient" and "Search for Patient" are presented in the window. (Figure B3M2)	With the two buttons "Create Patient" and "Search for Patient" the user has the possibility to create a new patient or to search for a pseudonym.

	5.	The user wanted to see if the patient is in the trial she entered a day before. She remembered that the name of the patient is "Peter". So, she clicked on the button "Search for Patient". *Comment:* "I cannot search because the pseudonym is not what I have actually captured in my data capture form on the desk." *Comment:* "I cannot go straight back." She clicked on the tab "Patients" and then on the sub item "Manage Patients".	A window is opened with an input field for the pseudonym of the patient (Figure B4M2). **Critical incident:** To start from the beginning and not going back step by step. The same window (Figure B2M2) is opened as before.	The user cannot search for the name of the fictive patient she entered. She has the pseudonym not on the data capture form on her desk, so she had to go back. She wanted to go back but didn't know how to go back. She didn't recognize the little sign ⌂ in the breadcrumb line. This sign is noted to return to the list of trials from this position. The sign should have a position where the user's attention is. It is usual to put the home sign at the first place in the breadcrumb list.
	6.	To see the expected patient the user clicked first on trial SIOP2001/GPOH.	The corresponding window with one entered pseudonym of the patient is opened. Two buttons are available "Create Patient" and "Search for Patient".	The user would like to see the patient with one click and not with two clicks. The patient has been entered in a trial. That's the point to open first the trial and then to go to the patient data. ➔ Suitability for the task The button "Create Patient" gave the user the possibility to enrol a new patient.
Subtask 3: Enrolling a new	7.	The user clicked on the button "Create Patient".	The new window presented an empty CRF for the new patient with a predefined pseudonym for the new	

fictive patient			patient.	
	9.	She entered a new fictive patient with first name "Lulu" and second name "Belle". Additionally the organisation is entered in the right corner. *Comment:* "This is my general patient."	The window presented the entered strings in the corresponding fields.	
	10.	She clicked on the button "Save". *Comment:* "Has it been saved? I got no message from the system."	**Critical incident:** No response by the system when saving.	The save action is not considered by the user. The user was confused. She was wondering if the save action has been happened already. There should be a message to tell the user the successful conducted action. ➔ Controllability, Suitability for the task, Self-descriptiveness
Subtask 4: Creating CRF for the new patient and looking for existing CRFs	11.	The user clicked on the tab "CRFs". She is wondering what to do now and is thinking about the philosophy of trials and CRFs. She decided to go to the trial she created some time ago.	A new window occurred with the button "Add CRFs" and the hint: "Please select a CRF to input data for this patient" (Figure B5M2). **Critical incident:** There is no CRF for selecting.	There is no CRF to select. The user is thinking what to do now. **Bug in the system!** Should be improved!
	12.	She clicked on the tab "Trials" in the main menu and then on the sub item "Manage Trials".	The existing trials are listed in the new window.(Figure B6M2)	It is the same window as in Figure B2M2 only with another name "View Trials"
	13.	She clicked on her created trial "Mariantesttrial"	The list of CRFs of trial Mariantesttrial was shown.	There is no breadcrumb to know from where are these CRFs. It would be helpful for the user to know the name of the trial if there will

				be various trials available. ➔ Self-descriptiveness
	14.	She clicked on the first created CRF named "Registration (F1)".	A new window occurred with a preview of the CRF's content under the trial Mariantesttrial.	The user was looking to enter more patient data into the trial but did not know where to conduct it.
	15.	She stopped the session because she could not enter more information for the patient		The session was stopped because the user could not finish her task in an efficient and satisfied way.

Occurred Usage Problems of both Tests of the first Prototype of ObTiMA

The occurred usage problems of both usability tests the data manager had during executing her task are listed in the following:

First test:

- without a training or introduction the user needs more information. She doesn't know which tab she can use to enter patient data
- consistency of the used language
- better comprehension of the menu items, there exists inconsistency to the tab names
- fields should be completely mouse sensitive and not only the string
- the "save" action is not visible for the user; she gets no acknowledgement
- she expected a CRF but no CRF could be selected → Bug of the system
- without response of the tool for saving, the user saved the CRF two times

Second test:

- user didn't know how to get her entered patient she enrolled the day before
- with two clicks the user reached her entered patient, one click too much
- "save" action not recognized in both tests because no response of the system
- the user runs again into the described loop of the first test → Bug of the system
- user wanted to enter more patients but didn't know how to conduct it

Recommendations regarding the dialogue principles ISO 9241-110

In the evaluation process the norm ISO 9241 (Ergonomics of human-system interaction) plays an essential role, especially part 110, the seven dialogue principles and part 12 the presentation of information.

The seven dialogue principles denoted by suitability for the task, self-descriptiveness, controllability, conformity with user expectation, error tolerance, suitability for learning and suitability for individualization concern the development of user interfaces and prevent users for producing usability problems. These could be

- unnecessary steps when conducting the task,
- incomprehensible information,
- insufficient and poor information on the user interface,
- unexpected and misunderstanding error messages,
- inefficient error recovery and
- navigation limitations.

A dialogue should be "the interaction (dialogue) between the user and the tool as a sequence of actions (inputs) and response of the system (outputs) in order to achieve a goal" where actions include additionally navigational and other control actions.

From this knowledge background the user needs in the first two tests more information how to apply a CRF to the enrolled patient data. It should be visible via mouse documentation or a previous introduction to handle CRFs in a running trial. For the navigation the user needs clear indication of tab names and the content of the website. If this is inconsistent the user is thinking about the action and does not know if the navigational action he/she conducted is correct. Data fields should be completely mouse sensitive because otherwise the user is wondering why no reaction of the system occurs. A very important feature the user is missing is the reaction of the tool for the "Save" operation. It is not directly clear that a short refresh of the tool denotes the "save" action. The user saves CRFs two times that has the side effect that it is time consuming and he/she had to delete one of the double saved forms. Thus deficiency has a high priority when improving the tool.

To reduce navigation steps can be an optimization of time effort, too.

The button "Add CRFs" irritates the user because there were no CRFs for selection. A recommendation could be to tell the user why no CRFs are there or inform the user in earlier steps to release a CRF for usage that could be selected when clicking on the button "Add CRFs". A description would be necessary for clarification.

The loop in which the user runs when allocating CRFs to the patient is a system bug and should be improved as soon as possible.

First test of the second prototype of ObTiMA
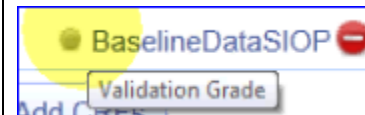
ObTiMA is tested by the data manager.

<table>
<tr><td colspan="5" align="center">Use Scenario – ObTiMA - MAT1

Screenshots in Appendix B, ObTiMA – MAT1</td></tr>
<tr>
<th>Task decomposed into subtasks</th>
<th>Step</th>
<th>User action and comment</th>
<th>Reaction of the dialogue system (ObTiMA)</th>
<th>Usage requirement, observed problems</th>
</tr>
<tr>
<td>Key task:

Enter patient data into the prepared SIOP trial which already exists in</td>
<td></td>
<td></td>
<td></td>
<td>Usage requirement:

To enter patient data in an efficient, effective and satisfied way (ISO 9241-11) into the Ontology based Trial Management Application (ObTiMA).</td>
</tr>
</table>

ObTiMA			**Initial situation:** The user is logged in correctly and the user starts with the list of existing trials which are shown in the window. (Figure B1MAT1)	We consider the dialogue principles of ISO 9241-110 and the information presentation of part 12
Subtask 1: Information about the SIOP trial	1.	User clicks on the SIOP2001/GPOH trial in the listing.	An overview of the SIOP2001/GPOH trial data is presented. (Figure B2MAT1)	The user gets full information about the trial.
	2.	In order to enter patient data she clicks under the main menu "Patient" on the item "Manage Patient".	A new window occurs with the pseudonym of the preexisting patients and the two buttons "Create Patient" and "Search for Patient". (Figure B3MAT1)	
Subtask 2: Create a new patient	3.	She clicks on the button "Create patient".	A new window with three columns for entering patient details occurs. The field of pseudonym is already filled out. (Figure B4MAT1)	The user is wondering if it is the patient she created some time before with this pseudonym. She has not the pseudonym in her brain.
Subtask 3: Enter the demographical patient data	4.	She enters the first and last fictive name "Peter Maier-Landruth" of the patient.	The entered names are directly shown in the corresponding fields on the screen. (Figure B5MAT1)	The user mentioned that it is a helpful option for entering date of birth.
	5.	She enters the country of birth "Germany" and with the selection button the date of birth.		

	6.	She enters the gender "Male" in the corresponding field.		To enter the language of the patient the user expected a better specification, e.g. does it mean mother language or the language of data input?
	7.	Entering the language the user is not sure what does the meaning of expression language be. She enters "English" as data entry language.	The chosen organization has a check mark. (Figure B5MAT1)	→ Missing Self-descriptiveness (ISO 9241-110)
	8.	The user chooses one of the organizations. Then she checks if all mandatory fields are filled out.		All mandatory fields have to be completed otherwise the user gets after "Save" a message for completion.
	9.	The user clicks on the button "Save".	A short refresh is visible. **Critical incident**: No response from the system. Figure B6MAT1 shows the patient entry without the pseudonym.	The user is wondering if the data entries are correctly saved. She gets no response by the system. → Suitability for the task (ISO 9241-110) **Recommendation**: The user needs a response for acknowledgement of the "Save" action. A refresh of the window is not enough. Another point is that the hint "Please select one or more organizations!" is shown after the correct selection of an organization (University of Oxford) and the "Save" action. (Figure B5MAT1) The hint irritates the user. After a correct save the hint should be deleted, so that the user is able to recognize the correct

				marking of the organization.
	10.	The user wanted to see an identifier to find the patient again in the patients list.		After the user has clicked on the "Save" button the pseudonym disappears on the screen

The user expected an identifier to find the new patient again in the patients list. When she entered the data the identifier exists and now she wanted to see the identifier as well again.

Recommendation:

The user should see the identifier as well as before saving the entry.

➔ Suitability for the task (ISO 9241-110) |
	11.	The user wanted to see the list of patients again. She clicks on button "Patient" and then on "Back to Patients List.	A new window appears with the patients list concerning the pseudonyms. (Figure B7MAT1)	She expected that her patient is the last one in the list. She does not have the pseudonym in her brain.
	12.	She guesses that the first pseudonym is her new patient she entered. She clicks on it.	A new window comes up with her currently entered patient data. But without the pseudonym of the patient.	The user misses the pseudonym. It is relevant to see the pseudonym as well as the name for further searching.
	13.	She clicks on the button "CRF" under the menu bar.	A new window comes up with the hint "Please select a CRF to input data for this patient" and the button "Add CRFs".	
	14.	The user clicks on the button	In the new window the "BaselineDataSIOP" CRF is	

		"Add CRFs" below	shown. (Figure B8MAT1)	
	15.	The user marks the CRF "BaselineDataSIOP" and clicks on the button "Assign selected CRFs to patient".	The check mark is shown directly on the screen. (Fig. 9) A new window appears with a red sign behind the "BaselineDataSIOP". (Figure B10MAT1)	The mouse cursor selects the CRF and the documentation shows "Validation Grade". The red sign with the mouse documentation are not described in any way.
	16:	The user mentioned: "The new screen tells me that the CRF for this patient already include BaelineDataSIOP."	**Critical incident:** The user needs more explanation for the mouse documentation and the red sign.	 The user needs more explanation here. It is not intuitive what to do in the next step because the user added a selected CRF to the patient in the step before. ➔ Suitability for the task (ISO 9241-110)
	17.	The user wanted to see the "BaelineDataSIOP". She clicks two times on it.	An exception message occurs. (Figure B11MAT1)	Failure from the system! This should be improved by the developers. The user doesn't know why this error message occurs.
	18.	She goes back with the browser button and clicked once more on the Baseline CRF.	A "Server Error!" occurs.	There was no more explanation. The user couldn't finish her task efficiently. She was unsatisfied with the work.
	19.	The user logged out.		

Second test of the second prototype of ObTiMA tested by the data manager

Use Scenario – ObTiMA - MAT2				
Screenshots in Appendix B – ObTiMA-MAT2				
Task decomposed into subtasks	**Step**	**User action and comment**	**Reaction of the dialogue system ObTiMA**	**Usage requirement, observed problems**
Key task: Enter patient data into the Ontology based Trial Management Application (ObTiMA)		The user lost her previous recording and tried the same a second time.		**Usage requirement:** To enter patient data in an efficient, effective and satisfied way into the Ontology based Trial Management Application (ObTiMA). We consider the dialogue principles ot ISO 9241-110.
Subtask 1:			**Initial situation:**	

To see the CRFs for existing patient.			The user starts with patient data of the fictive patient "Peter Maier Landroth". (Figure B1MAT2)	
	1.	The user wants to see the CRFs for patient "Peter Maier Landroth" and clicks on the item "CRF" in the menu bar. *Comment:* "There are no records suitable for this patient" which I can select from to attach. I would expect a list of CRFs relevant for the trial."	A new window is opened with the message "No records found". (Figure B1MAT2a)	The user expected a list of CRFs that are relevant for the trial. → Conformity with user expectations (ISO 9241-110)
	2.	The user goes back to the Patient List via the breadcrumb.	A list of three patients is presented with their pseudonyms in a new window.	
	3.	She selects the third identifier in the list and hopes that it is the patient she entered before.	No reaction by the system!	
	4.	Instead of waiting she clicked on the button "Search for Patient".	**Critical incident:** The user is waiting for a reaction but nothing happens.	That is not efficient to wait and nothing happens. It has to be checked by the developers.

			A new window with the field for "Pseudonym" appears.	One reason could be that the user clicked in the identifier field but not directly on the identifier. This could be the problem for long waiting.
	5.	The user enters the identifier she has in her brain to get the data more quickly and press the button "Search for Patient".		**Recommendation:** The whole field should be mouse sensitive and not only the identifier if it is a short identifier. A very long identifier is also not so useful because of producing typing errors.
			The new window is shown with the patient details quite quickly.	The user needs a response of the reason by the system.
Subtask 2: To get the CRFs for the patient	6.	She wanted to see the CRFs for this patient and clicks on the item "CRF". *Comment:* "I got stuck again. A few minutes ago with the failed recording I was able to see the CRFs."	A new window appears with the hint "You do not have the rights to see or add CRFs." (Figure B3MAT2)	The user does not understand why she cannot see the CRFs which she saw a few minutes before in the lost recording. The user should have the same rights for her entered data. She has the full responsibility.
	7.	She goes via the breadcrumb back to "Patient List" and selects the pseudonym of her patient in		

		the list again.	The patient details are shown.	→ inconsistency
	8.	She clicks on the item "CRF". *Comment:* "Now, I can see the CRFs when I go over the patient listing but not via the search function. There is an inconsistency."	Now the CRFs are listed on the screen **Critical incident:** The user cannot see the CRFs when she uses the search function for patients. But when she goes over the "Patient pseudonym List" she sees the expected CRFs.	This is an inconsistency of the search function and the patient listing. The user gets different sights. The user should see in both cases the same result. → Suitability for the task → Conformity with user expectations
Subtask 3: To see the sections of the trial.	9.	The user clicks on the first CRF. *Comment:* "The clinic number should be entered in form of an integer and not as a decimal."	The full content (entered before) of the CRF with section demographics is shown. (Figure B4MAT2)	Some time ago the user has entered the full name of the patient, the patient number, the clinic number and the other relevant identification numbers. The user wanted to have an integer for the clinic instead of a decimal. → Should be improved.
	10.	The user changes the number of the clinic because of typing error with the edit function. She enters also the reason for changing the value.	All changes are seen on the screen.	It's a nice feature for the user.
	11.	She moves to the next section "Operation Details" via the	A new window of the BaselineDataSIOP trial appears with operation details. (Figure	The user is wondering that when she saw it the last time there were more fields on the screen. She wanted to edit the fields

		scrolling signs. *Comment:* "I'm not sure what the red colour code means. In other trial systems it means that there is an error or query; I think it means something different here."	B5MAT2)	as she did it in the demographics section but there was no possibility to add some information. The red circle needs to be described. It is inefficient to think about such things and got no answer. A mouse documentation could be helpful. The user does not know why she is unable to edit the entries in this section.
	12.	The user wanted to add a new CRF. She goes back via the breadcrumb to her patient.	The list of the CRFs is shown again.	
	13.	She clicks on the button "Add CRF" at the left bottom. Comment: "I could not add a new CRF for the patient, because the one allocated trial CRF had already been assigned to the patient!"	The same window is shown as in Figure B1MAT2a with the message: No records are found.	The user runs into the same loop as in step 1. She wanted to add a new CRF but she is not able to handle it in an easy way.
	14.	The user clicks in the main menu on the sub item "Manage Trials".	Two running trials are shown in the new window. (Figure B6MAT2)	
	15.	She selects the second one named "SIOP2001/GPOH trial". *Comment:* "This trial is running."	Detailed information of the trial is presented with the menu items "CRFs" and "Biobanks".	

	16.	She clicks on the menu item "CRFs". *Comment:* "There is only one CRF allocated in the trial."	A new window occurs with two CRFs of different status and the two buttons on the bottom "Create CRF" and "Add CRF from Repository". (Figure B7MAT2)	
	17.	The user clicks on the button "Add from Repository".	A list of CRFs from Repository is shown. (Figure B8MAT2)	
	18.	She selected the CRF for Chemotherapy in the list.	A window is opened with the Chemotherapy Group as a preview.	The user sees that it is a CRF from another person and she wanted to assign a CRF to her patient but she has no idea to manage it. It is not possible for the user to conduct her task efficiently.
	19.	She stops her session unsatisfied.		

Third test of the second prototype of ObTiMA tested by the data manager

		Use Scenario – ObTiMA – MAT3		
		Screenshots in Appendix B – ObTiMA-MAT3		
Task decomposed into subtasks	**Step**	**User action and comment**	**Reaction of the dialogue system ObTiMA**	**Usage requirement, observed problems**
Key task: Enter patient data into the Ontology based Trial Management Application (ObTiMA)				**Usage requirement:** To enter patient data in an efficient, effective and satisfied way into the Ontology based Trial Management Application (ObTiMA). We consider the dialogue principles of ISO 9241-110. From the view of a data manager the task was also a data manager task of trial design: Allocating CRFs relevant to the needs of the study (under GCP for CTIMPs.

				This done is a very controlled and quality managed way. To capture data items in the trial protocol, no more no less.)
Subtask 1: To allocate CRFs for the previously entered patient	1. 2.	The user logs again into ObTiMA with her user account via the url https://obtima.org/test/development-snapshot/ . She was thinking why the CRFs she entered before are not available for the user. *Comment:* "I didn't release them. This could be the reason why the CRF is not available for my patient. I have to check this." She clicks on the menu item "Trials" and selects the sub menu item "Manage Trials".	**Initial situation:** The website of ObTiMA to login is shown. (Figure B1MAT3) Two existing running trials are shown in the new window. (Figure B2MAT3)	When the user allocated CRFs she forgot to release them. The user needs a clarification or a hint from the system. At least the user found out the reason by herself. **Recommendation:** A hint could notice the user not to forget to release the CRF.
	3.	The user clicks on the trial named	The trial information is	

		"SIOP2001/GPOH"	presented.	
	4.	She selects from the menu the entry "CRF".	A list of CRFs is shown.	Only one CRF the "BaselineDataSIOP" is released. The other CRFs have to be released, noted in the status column.. It is essential necessary in the design to have as a header on the screen, the name of the present trial to which the user allocates CRFs during the whole session.
	5.	The user selects the "CRF for Chemotherapy" and clicks on the status symbol.	Mouse documentation is presented to tell the user to release this trial for usage within the trial. (Figure B3MAT3) Then a pop-up window appears for releasing. (Figure B4MAT3)	
	6.	The user clicks on the "OK" button. *Comment:* "The system doesn't leave me where I'm." "I wanted to stay on that screen with the CRFs' list, so that I could review it again before leaving the screen."	The trial information is shown once more. **Critical incident:** The user wanted to see the released CRF for her acknowledgement. The system does not leave her where she was.	This is not what the user wanted to see. She is thinking about it. She wanted to see the released CRF? When she did the same thing at step 24, it DID stay on the CRF list screen. The developers have to check this inconsistency. ➔ Suitability for the task

				→ Conformity with user expectations
	7.	She goes back to the trial. The user released a second CRF "Adverse Event Form". She has to go a second time over the overview of the trial to the released CRFs. *Comment:* "I was SENT back to the trial overview (when I would have preferred to stay on the CRF list screen)."	The same occurs as before. The marked CRFs are shown. (Figure B5MAT3)	The user is not able to see the released CRF directly when clicking on the item "CRF" under the trial. She got an overview of the trial, but that is not what she expected. The user has to go back. It is not efficient.
	8.	The user wanted to release a third CRF "Pathology (F4)" and clicks on the status. *Comment:* "I can't use it." "This is good because the system is telling me that the CRF selected is still under production, not ready to be used."	A pop-up window occurs that tells the user that a question has to be entered before releasing. (Figure B6MAT3)	The user closes the information window and goes back.
	9.	She selects the item "Patient" and then the submenu "Manage Patients". *Comment:* "I hope when I go back to the patients of this trial I will be able to use one of my released CRFs for them."	A list of three patients' pseudonyms are shown. (Figure B7MAT3)	The patient data are shown very quick in contrast to the last test. The reason

	10.	She clicks on one patient pseudonym.	The patient data are shown.	can be the click on the identifier and not as in the previous test on the field.
				This should be improved. The whole field has to be mouse sensitive.
	11.	The user clicks on the item "CRF" above the patient data.	The available CRFs are shown with the button "Add CRFs" at the bottom.	A long list of the same CRF is shown. It should be possible by the user to remove this failed input.
				→ISO 9241-110: Controllability and Suitability for the task
	12.	She clicks on the button "Add CRFs" and marks both CRFs "CRF for Chemotherapy" and "Adverse Event Form"..	The two new released CRFs are shown in a new window. (Figure B8MAT3) with a check mark.	That is what the user expected. She is satisfied.
	13.	She clicks on the button "Assign Selected CRFs to Patient".	The marked CRFs are directly presented at the end of the already existing CRFs.	Now the user achieved what was impossible in her previous tests.
			It shows the title, version number, a description and day /time of the last modification.	
	14.	She selects the "Chemotherapy CRF" in order to enter data of her patient.	The details "CRF for Chemotherapy" is presented with empty fields for entering data. (Figure B9MAT3)	A typing error in the expression "**Chemotherapie** Group"
				This should be improved.
				It is important in the design to have as a header on the screen, the identifier of the patient for whom users are inputting data.
	15.	After correctly entered the date of birth	No value for T and N	This must be a mistake by the system

		and the histopathological grade the user wanted to enter via the option list the T and N classification and the pCR value. *Comment:* "There is no further section for Chemotherapy where I can enter data."	classification is shown. **Critical incident:** The user is not able to enter the value for T classification and for N classification, too. (Figure B10MAT3) No reaction from the system.	and should be improved. The problem was that, though the drop-down menu is a good design, she could not input data; she is thinking this is probably a just small programming bug (maybe the field input width is too small?), the concept is good. Also the feature of conditional questions is good. If the answer to "AE stop date known" is "Yes", then a new field appears to capture the actual date. The user has to omit the relevant values. The opinion of the user is that the data items on the screen were good and typical and probably appropriate to the study in question.
	16.	She clicks on the button "Save". *Comment:* "Oh, it seems it has been done."	A short refresh of the window is visible. The date of birth disappears and at the right side of the window small red buttons are visible for the entered data. The previous entered data are gray out. (Figure B11MAT3)	There is no acknowledgement for the user that the save was correctly executed. Gray out means that it cannot currently selected by the user. The user needs some explanation, especially for the red dots at the right side. Red is not the usual colour for correct execution. Green would be a better colour. Comment from the user:

				"Yes, green to indicate successfully input, would be good (in our commonly used system; red indicates that a query has been raised to check the data." **Recommendation:** A legend for the various colours should be presented.
	17.	The user clicks on the breadcrumb "Back to Patient".	The list of CRFs is shown.	The breadcrumb "Back to Patient" associates not to go to the list of CRFs. It should be better renamed by e.g. "Back to Patient's CRF list". The present name rather relates to the list of patients. New users can be getting confused because of misunderstanding. ➔ Self-descriptiveness
		She selects the "Adverse Event Form" for entering patient data as previously. She enters the data in both sections and saves the entered values. *Comment:* "In practise patients will get several Adverse Events (AEs), so that I will allocate another AE CRF to the same patient to put in a second event."	The reaction from the system is similar to the activities described for the "CRF for Chemotherapy" (s. step 14).	Everything works well. The user is able to enter all relevant data into the available fields. Only from the "Save" operation the user receives no response. She hopes that the save operation was successful.
Subtask 2:				The user wanted to create a new CRF

To add one more CRF for an Adverse Event Form	18.	She decides to create a second "Adverse Event Form" for the same patient. So she returns to the CRF list for her patient via the breadcrumb "Back to Patient".	The list of CRFs is shown again with the button "Add CRFs".	for the Adverse Event Form. In practise patients can have more than one adverse event forms.
	19.	The user clicks on the button "Add CRFs"	A new window appears with the information "No records found". (Figure B12MAT3) **Critical incident:** The user gets not enough information to know how to add a copy of the existing CRF to her patient.	The message the user receives from the system is not understandable. The user needs more information to conduct the next step efficiently. ➔ Self-descriptiveness ➔ Suitability for the task
	20.	She goes back to her patient. *Comment:* "For my patient I would like to add another adverse event as I had. It should be a copy of the first adverse event form for e.g. repeat blood results, repeat restaging of the tumour and recording items at each cycle of treatment. The CRFs could be reusable. But even better, it would be an idea, if I teach stage of a clinical trial all of the relevant CRFs at that stage according to the approved protocol would be bundled together into a visit package or event package or whatever you want to call it. So each CRF is used		The user looks into the repository and selected another CRF that she added in a correct way to her patient but it is not what she wanted. The designer has to think about the user's idea to enable the required functionality. ➔ New requirement has to be considered.

		once on a visit but it is grouped together with everything else that is relevant to the same visit or event."		
Subtask 3: To allocate another CRF from the repository to the patient	21. 22.	The user goes via the menu item "Patient" to the list of CRFs. She clicks on the button "Add CRF from Repository". She selects another CRF out of the repository, the "IMPORT - Post-operative treatment (F6)"	The list of CRFs of the patient if shown with the button "Add CRF from Repository". The CRF IMPORT... with empty fields occurs	
	23.	The user clicks on the button "Add CRF to Trial".	The system shows the existing CRF list with the new added CRF for the trial.	The user is thinking about the following when she goes to the "CRF" for a patient she sees the list of CRFs that have been assigned to the trial but have not yet been assigned to the individual patient. It is important for the user to see the list of CRFs for the patient.
	24.	The user clicks on the release sign of the new IMORT... CRF. Comment: "At this point I stayed on the trial's CRF list. This is what I wanted. However it did not happen like this when I was in a similar situation in step 7."	The CRF is marked by a check mark	In this case the user receives an acknowledgement from the system that the CRF is released. That is a useful feature.
	25.	Now the user is hoping that when she goes to the patient that the new CRF is released and can be added to the	The list of all available CRFs without the IMPORT... CRF is listed.	The naming of the breadcrumbs is not consistent to the visible content of the window.

		patient. The user clicks on the button "Add CRFs". *Comment:* "That is fine. I can put this in. But the same CRF will be needed and repeat blood results, repeat restaging of the tumour and recording items of each cycle of treatment."	A new window with the new CRF is shown. A hint is given to the user "Please select one or more CRFs to assign to patient". (Figure B13MAT3)	It should be adapted, so that it presents a better understanding for the user. The user noticed the same as in step 20. She has no idea to get a second Adverse Event Form.
	26.	Anyway from now she clicks on the button "Assign Selected CRFs to Patient".		
	27. 28.	Then the user fills in the empty fields with relevant values in both sections "Post-operative Chemotherapy" and "Radiotherapy". *Comment:* "That's good. The questions are fine. There is specific for the patient who is treated." The user stopped the session.	The forms are shown and the edited fields are presented.	Everything works well and therefore we need no description. At the end the user stopped the session. The lack of a couple of same (reusable) CRFs is her great problem.

Occurred Usage Problems of the second Prototype of ObTiMA

We collected the occurred usage problems of all three usability tests of the data manager in the following:

- still missing save acknowledgement
- failing in expecting a list of CRFs for selection
- inconsistency of the search function to the tab "Patient List" when looking for CRFs of the patient
- no rights for her entered patient ← system error
- the "Patient" can be seen via "Patient Pseudonym"
- clinic number should be entered in form of an integer and not as a decimal
- the whole identifier field should be mouse sensitive and not only the identifier itself
- red circles in the description of CRFs are unclear for the user. It is not usual to use a red sign for 'accepted'
- the hint of marking an organization should be deleted after selecting and marking correctly
- error messages should be readable and comprehensible for the user
- very long identifiers causes typing errors
- inconsistency; different views of patients
- to apply a new CRF to the entered patient is not possible ← system error
- when the user allocated CRFs she forgot to "release" them and therefore they were not available for the user
- missing header of the trial during the whole session
- if releasing a CRF the user wants to see the released trial for reviewing again
- going back to the trial overview is inefficient
- typing error in the expression "Chemotherpie"
- header of the screen should contain the identifier of the patient to whom users are enrolling data
- colour clarification is necessary
- better naming of breadcrumb items, e.g. "Back to patient" could be better in "Back to patient's CRF list"
- saving the entered values, results in no response of the system; the user can only hope of successful saving
- unable to use an "Adverse Event Form" a second time, the user needs a copy of the first adverse event form
- the message "No records found" is not comprehensible for the user
- consistency of the breadcrumbs to the content of the window

Recommendations considering the dialogue principles (ISO 9241-110):

In the three usability tests which were conducted with the second improved prototype of ObTiMA we recognized that some required functions or features have still not yet been implemented.

In the meantime the user learned to "release" CRFs after completely filled out with patient data. This causes a correct apply of the CRF to the corresponding patient.

From the presentation of information (ISO 9241-12) it is essentially necessary that the user has all relevant information presented on the working window during his/her session. The name of the trial the user is working with should be visible on the window. For CRFs the identifier of the patient the user is editing should be visible in the header of the screen.

The red sign after filling out a CRF is not a usual sign for completion.

A recommendation could be to show a legend with the different colours and description for better understanding. It is an essential need because otherwise users are thinking about the colour that the input could be wrong. This will be discussed in the developers group.

From a developer we got the following answer:

There exist two levels of validation for a patient CRF:

- one for each data field in a CRF and
- one for the whole CRF,

Initially all fields in a CRF are grey marked because they have no contents. When a data field is filled out by a data manager, a physician or imported automatically, then the data fields become red because of missing validation by the trial manager. Only the trial manager is able to change the status of the field after control to yellow if there is further discussion about it or green if it is correct.

In any case it is important to have comprehensible error messages, so that the system assists the user in detecting and avoiding errors. An error message should be provided to the user to facilitate the correction of the error.

All coloured graphical symbols should be explained.

Breadcrumb items should be conformed to the content of the window without confusing the user's expectation.

The navigation should be in this sense that the user knows in any place where he/she is, from where he came and what the next step will be.

The user needs more explanation, especially in handling "Adverse Event Forms" a second time. There was a field to mark a CRF as "Repeatable" that the user didn't know what it means. It denotes to use the "Adverse Event Form" twice or more.

A short introduction should be available to support and guide new users in learning to use the tool in an efficient, effective and satisfied way. This concerns either the menu items as the fields to be filled out. For the heterogeneous groups of users this means to have different support needs and more or less explanations which should be optional.

4   The Ontology Annotator (OA)

The Ontology Annotator (OA) is a web-based graphical tool targeted at p-medicine end users and that allows manually defining annotations for external databases. These annotations will allow the subsequent integration of the annotated database into the p-medicine infrastructure.

By annotation, what we mean is the alignment of the database elements with HDOT (the ontology libraries used within p-medicine). The idea is that the users employ the OA to establish relations between semantically equivalent terms of the annotated database and the ontology libraries. The HDOT ontology can be seen, in fact, as another database schema, used to represent the integrated data within p-medicine. The OA facilitates this task by providing graphical representations of both the database to annotate and the HDOT ontology. Users are then requested to select elements from each of the schemas to define the database annotation. Figure UPM1 shows the main OA window.
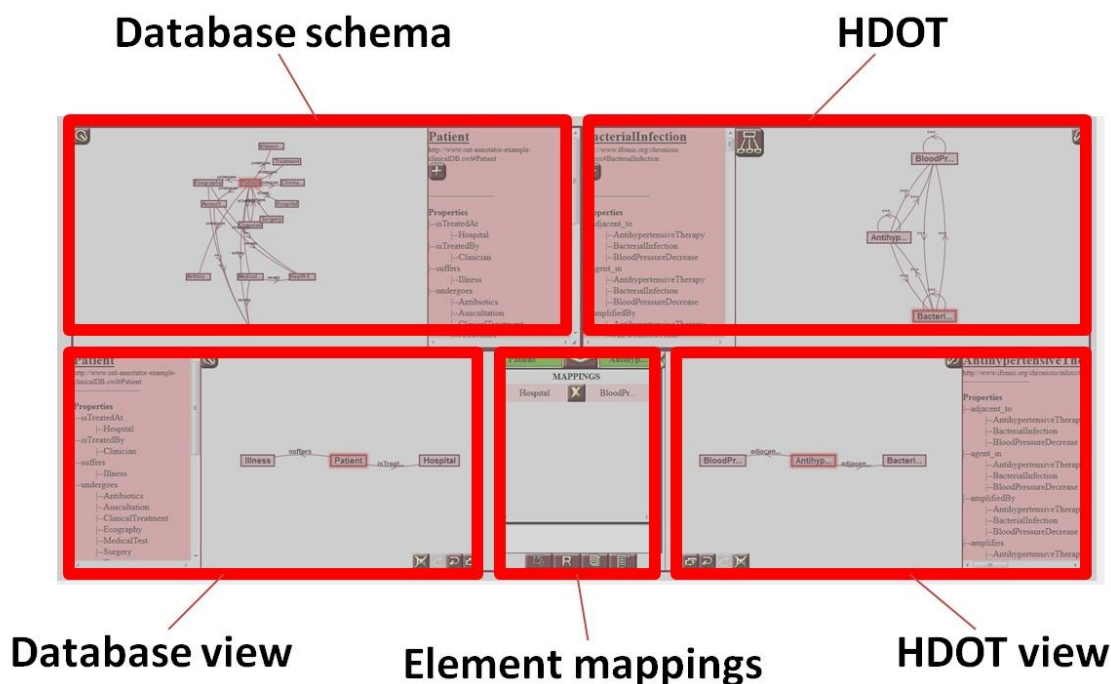
Figure 4:   the main OA window and the canvases that compose it. Top left canvas displays the database to be annotated. Top right canvas displays the elements of the HDOT ontology. Bottom left and bottom right canvases display, respectively, one database view and one HDOT view built by the user. Bottom middle canvas displays the individual element mappings between both views.

As it can be seen, the main window is divided in several canvases, each providing graphical representation of one element in the annotation process. Both the database and HDOT are

represented using graph-like structures. Users are expected to be able to navigate through these graphs in order to find semantically related concepts that must be aligned. In the case of HDOT, users must first select the elements of this ontology that they wish to work with. HDOT elements are grouped in modules to facilitate this task, and users can select the modules of their interest (e.g. "clinical module", or "DICOM module"). The selection of the appropriate HDOT elements is performed by searching for the element names in a text box.

The annotation process consists on defining pairs of views (one from the database and one from HDOT) that express the same concept, and then establishing mappings between the individual elements of the two views. This implies navigating through the HDOT structure and searching for adequate elements that provide the same meaning as the database elements.

Usability Test for the Ontology Annotator

The usability tests carried out for the OA have focused on the user interaction of the tool. The goal was to evaluate how efficiently users were able to use the features provided by the OA. More specifically, the designed tests evaluated the following areas of use:

1) Project creation: the user should be able to create a new annotation project, selecting an example database.
2) Database visualization: the user should be able to easily visualize the components of the example database.
3) HDOT navigation/configuration: the user should be able to select the appropriate modules of HDOT, and to load the elements necessary for the experiment.
4) View creation: the user should be able to create views both for the database and for HDOT.
5) Individual mappings creation: the user should be able to generate the necessary mappings between elements of two views.
6) Entry saving and navigation between existing entries: the user should be able to easily use the controls for saving
7) Project saving and submission: the user should be able to finalize the working project, and submit it to the p-medicine framework.

Following these topics, a detailed usability test plan was prepared for users. The users were provided with a tutorial of the tool, and then they were given instructions on the tasks they had to complete. The task descriptions only provided general tips, as the degree on completion of each of them would subsequently be used to evaluate to what extent the users had learned to use the OA tool just by reading the tutorial, and in which aspects they had more trouble. The OA tutorial can be found in Appendix D, while Appendix E presents the usability test with each of the tasks to complete.

Usability tests with the first prototype of the Ontology Annotator

The first usability tests were conducted by members of p-medicine. They detected some first deficiencies concerning the suitability of the task (ISO 9241-110) and the presentation of information (ISO 9241-12) that where communicated to the developers for improving the software.

To these issues a main problem was that five buttons "SUBMIT HELP EditUsers EditDescription EXIT" were presented in a menu in form that the content is invisible during the session. It comes only up when mouse moving over it. Figure 5 shows this effect of hidden buttons (s. blue arrow).

If the buttons could be placed on top of the window in form of one line the user is aware of them during the execution of his/her task. That is necessary especially for the "Help" button which could be described on a separated page with an intuitive documentation.
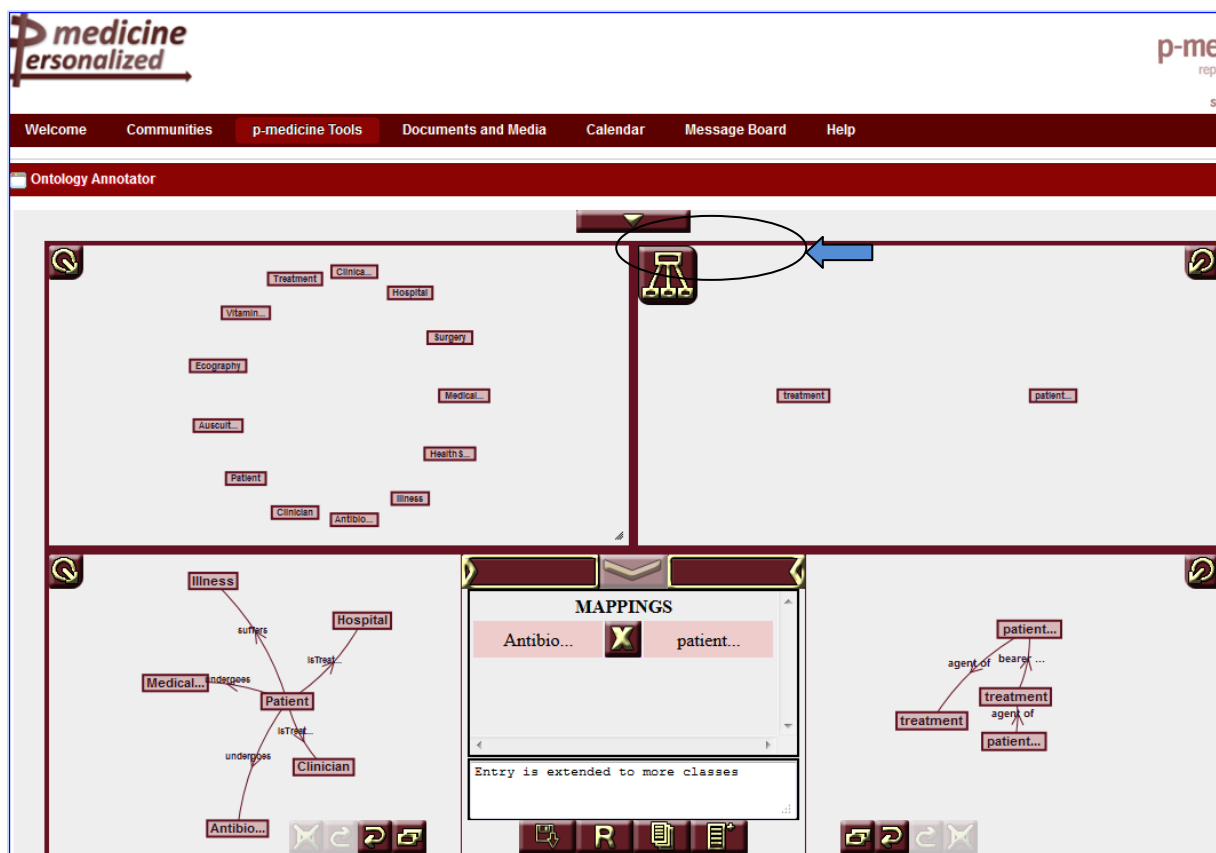
Figure 5: Ontology Annotator website with hidden buttons on top

Another point was when an entry was saved it would be good to ask the user for an entry name. To close the window it would be better to have an icon and a message "esc or exit" instead of "cancel".

All these issues result in improvements of the OA and flow into the second prototype which was tested by an external user on the "Summer school for computational oncology". The use scenario is listed in the following:

Use Scenario from an external user

		Use Scenario – Ontology Annotator - SS13		
		Screenshots in Appendix C – SS13		
Task decomposed into subtasks	**Step**	**User action and comment**	**Reaction of the dialogue system Ontology Annotator**	**Usage requirement, observed problems**
Key task: Annotation of one example database that is quite simple as it provides just a few attributes about patients of a clinical				**Usage requirement:** The Ontology Annotator (OA) is a web-based graphical tool targeted at p-medicine end-users. It allows manually defining annotations for a wide array of various biomedical databases (including clinical data and genetic repositories). These annotations allow subsequent integration of the annotated database into

				the p-medicine infrastructure.
study.				First the user has to read the short introduction of the OA and then follow the description of the task on paper.
Subtask 1: Create a new project in the OA.			**Initial situation:**	Problem was the registration. Therefore the usability engineer logged in with her account.
				The problem has been solved in the meantime!
			The user is logged in under the account of the usability engineer.	The screen should be better used for the graphical presentation of the databases.
	1.	The user clicks on the '+'sign.	The reference email of the usability engineer and not the email of the user is shown and with click on the '+'-sign it denotes to create a new project.	**Recommendation:** The long written description on the right side should be clickable like a button If the user needs this help he/she can choose to open it.
			A window appears for entering the name of the project, its description and the type of the database. (Figure C1)	➔ ISO 9241- part 12 Presentation of information
				This produces more place for the main graphical content.
	2.	The user follows the written task description on paper. He entered the name of the project in the first field, "Example exercise annotation", the description "An annotation		The test was stopped here in order to read the introduction first by the user before conducting the task.
				Without reading the short introduction the user is not able to conduct the described task efficiently.

		example for the usability tests".		
		Because of not reading the introduction first the user is unable to click directly on the correct option of database type.		
		The user continues the session with the "Example exercise" database after he read the introduction in detail..		
Subtask 2: Select the appropriate ontology library modules.			Figure C2 shows the exercise database in the database schema canvas with selecting class "patient" and in the HDOT canvas the opened menu for defining the used modules.	After reading the introduction the user continues his task.
	3.	The user clicks on the item "Modules" in the HDOT canvas menu.	A new window appears for selecting the appropriate modules (Figure C3).	
	4.	The user needs some time to get them into his focus and then marks both "HDOT KLt" and "HDOT DICOM" modules	The marked modules are shown in the window.	To see the relevant modules for selecting it would be more efficient for the user to arrange them in alphabetical order.
	5.	He clicks on the "ok" button.	The window disappears.	The user gets no acknowledgement about

			Critical incident:	correct saving of the selected modules.
			No acknowledgement of the correct saving.	A short information about the marked modules could also be assure him for correct marking.
Subtask 3: Add the classes from HDOT; Search for equivalent classes for "Patient"	6 7..	He enters in the "Search" field the name "patient" and clicks on the "Search" button. He scrolls along the list of classes. Finally he finds the class "patient" and clicks on it.	A list of classes in the HDOT canvas are presented. The field of class "patient" becomes green and in the HDOT window the class appears. (Figure C4) **Critical incident:** The class is partly under the menu. If it would be completely hidden the user will not see it.	The user goes with the mouse over them but doesn't know which one to select. A short mouse documentation gives him more information. After clicking on the class he expected the selected class in the window on the right but it disappears partly under the menu and in other cases it is completely hidden under the opened menu. The user is looking for the class and it is waste of time to work efficiently. ➔ ISO 9241 – 110 Dialogue principles: Suitability for the task, Controllability, Conformity with user expectations Hint from the user to put the parents and children in one place to get more place on the screen for the main graphical content. **Recommendation:** To enlarge the window in reducing the documentation on the right side of the screen that is not necessary when conducting the task.
Entering class "identifier"	8.	He enters into the "Search" field the name "identifier" and clicks	The corresponding list of classes is shown in the HDOT canvas.	

		on the "Search" button.		
	9.	The user selects the class "identifier" out of the presenting list of classes.	The field with entry "identifier" becomes green and the class appears in the main menu on the right but the class patient is hidden. (Figure C5) **Critical inciden**t: The same happened as described in step 7.	The user doesn't know how to get class patient back. It was hidden behind the menu.
Entering class "biopsy"	10.	The user enters the name "biopsy" in the "Search" field and clicks on the "ok" button.	Nothing happens, only the line "Search ("biopsy")" appears. (Figure C6) **Critical incident:** After some seconds the list of expected classes occurs and the string "Search ("")" is empty. It shows the user that the system finished the search action, but that is not what the user recognizes.	The user doesn't recognize that with the line "Search ("biopsy")" the system informs the user to wait for the respond because the search action has not yet finished. It is not intuitive for the user that he has to wait until the system has found the appropriate class "biopsy". If the system has achieved the result the string "Search ("")" is empty. This should be better denoted by the developer, e.g. please wait until searching is finished.
	11.	The user scrolls though the long appearing list of classes but doesn't find a useful name. He was thinking if it is the correct list of biopsy classes..		If the list of classes were in an alphabetical order it would be easier to find the expected class. The user was irritated and tries another name "suspect tissue". He doesn't recognize that the search for "biopsy" has not yet finished..

Entering class "suspect_ tissue"	12.	He tries another expression he enters "suspect_tissue" into the "Search" field.	The name occurs in the "Search" field but the line below shows the user that the search action for "biopsy" has not yet finished. (Figure C7)	This shows that the user did not recognize the important line "Search ("biopsy")". ➔ ISO 9241- 110 dialogue principles ➔ ISO 9241 – 12 presentation of information
	13.	He clicks on the "Search" button. *Comment:* "The searching is not correctly working."	The "Search" field becomes empty but the line with the expression below "Search ("suspect_tissue")" shows now the new searching expression. (Figure C8) **Critical incident:** The same as described before in step 10.	The user doesn't know what to do. He goes back to the list of modules and was wondering if he chooses the correct modules. He did not. He chooses HDOT DICOM and that could be the reason why the system doesn't finish the search action of "biopsy". The developers have to check.
	14.	He goes back to the list of modules. Then he saw that he marked "HDOT DICOM". He deselected it and marks "HDOT Core" and closes the window.	The window of modules is appearing.	
	15.	He tries again entering "biopsy".	The list of appropriate classes in the HDOT database does not appear.	The user has been at a loss. He doesn't know what to do.
	16.	He enters a new expression "neoplasm" into the "Search" field and clicks on "search".	A new list of appropriate classes appears in the HDOT canvas.	
	17.	He chooses the appropriate	The selected field changes the	The user misses the data entry in the

		class "neoplasm".	colour to green.	main HDOT canvas.
				The same situation as described in step 7.
				The developers have to think about an improvement to make the selected classes visible and not hidden.
	18.	The user enters another expression "surgical procedure" and clicks on the "Search" button.	One entry is shown.	The same procedure as before.
	19.	He selects the shown entry "surgical procedure"	The field becomes green but the class is not visible in the window.	The user has to minimize the classes to see the hidden ones.
				There are only four classes which should be visible without zooming.
				This is not efficient.
Subtask 4: Create a view of the database which represents the concept "Patient identifiers".	20.	The user clicks on the class identifier in the database schema and add it to the database view.	Two classes are shown in the database view.	The class patient has already been added to the database view.
	21.	He clicks on class patient in the database view and adds a relation "has identifier".	The relation is shown in the window	A side effect of the relation "patient has an identifier" is that the system produces a new class identifier one standalone class "identifier" exists which has to be deleted by the user.
Subtask 5: Create a view of the database	22.	He tries to do the same relation in the HDOT view but with some problems.	The relation is shown in the window **Critical incident:**	The user has some trouble with finding the corresponding property in the HDOT view. The window is too small to read the

from the ontology libraries.			The property for the relation is not easy to find in the list because the window is not large enough to show all information without scrolling left / right and top / down. (Figure C9)	names in an efficient way. The user has to scroll many times up and down and left and right to find the property he wanted.
				➔ ISO 9241 – 110 Dialogue principles : Suitability for the task
				➔ ISO 9241 – 12 presentation of information
Subtask 6: Establish individual mappings between the elements of the created views.	23.	The user clicks on the class patient in the database view and mapped it to the window in the middle, the element mappings.	Class patient is seen on top of window "Mappings" and the field has the colour green.	
	24.	Then he selects the class patient in the HDOT view and maps it to the mappings window, too.	Class patient of the HDOT view is also visible in the "Mappings" window on the right side with colour green.	Both classes patient from the database view and from the HDOT view have the same meaning and can be mapped therefore.
	25.	Then the user clicks on the "add map" button in the middle of the window.		Requirement of the user is to use standard symbols on the windows.
		Comment: "The symbols used on the windows should be standard."	Both classes patient as a link are shown in the "Mappings" window. (Figure C10)	Here the session is stopped.

Occurred usage problems of the second prototype of the Ontology Annotator

The Ontology Annotator (OA) was tested on the Summer School of Computational Oncology in Dagstuhl (Germany) from an external user. He saw the user interface and the task for the first time. This situation shows also initial usage problems which occurred when users are not acquainted with the tool.

The following usage problems can be listed:

- registration doesn't work at the first time an external user tried it
- the whole screen should be better used for the important content
- the description about handling the tool is necessary to work efficiently with it
- difficult to search for modules under HDOT; better to arrange them in alphabetical order
- if clicking on the "ok" button the user gets no acknowledgement about the successful saving
- missing mouse documentation for better recognizing the relevant classes of HDOT
- arranging the graphical part of the window, so that all selected classes are visible and not hidden under the menu
- in order to get more place for the graphical output to put parents and children in one place ← hint of the user
- message of the search function is not intuitively understandable
- the tool should inform the user about the situation to wait until the search action finishes
- alphabetical order of the list of classes and the properties
- better graphical presentation; menu window of the properties too small for efficient reading
- to use standard symbols on the window

Result of the usability test

The validated use scenario was sent to the developers to take into consideration the problems the user had to conduct the task in an efficient and effective way. Some of the detected problems with answers from the developers were displayed in the following table. It shows that the test has been very fruitful in the sense to improve these detected issues and develop an improved version for the next usability test.

Table 1: issues encountered during the usability tests, and the corresponding actions to be taken.

Issue	Answer
Portal screen uses to much space for the tool description. It should be worked on	This has to actually do with the Portal implementation and the way they access the OA tool. Discussions between the OA development team and the p-medicine portal developers will take place in order to optimize

	the visualization area of the OA, removing unnecessary parts (e.g. moving the OA description to a separate window).
HDOT module pop-up window is very unusable	Agreement to work on improving the HDOT module selection window was reached. Yes, I agree this must be improved. In the first place, the displaying of the available modules will be more compact and descriptive, and in alphabetical order. In the second place, a confirmation message that informs the user which modules have been activated will be included.
Tool is unusable with small resolutions. This provokes several problems, as the hiding of classes in the HDOT window, of the need for scrolling from left to right when searching for a suitable relation in a view window.	Make smaller buttons. Discuss about the possibility of providing two tool modes: one for large screens and one for small screens that reduces the font size and images Point out the ability to enlarge windows in the tutorial.
Search results are hard to review	Results will be displayed in alphabetical order, and HDOT element searches will be dynamic so the user gets better feedback of his actions.
Non-standard button images	Replace button images with more standard icons.

The registration problem was solved in the meantime but it has not been tested again by an end-user. This should be done in the next project stage.

The main content of the window should take the most place of the window because that is what the user needs when conducting the task. The table above displays the main problems with first solutions or discussions of the developers about possible improvements.

5 Results of usability tests from participants of the Summer School of Computational Oncology

The participants of the Summer School for Computational Oncology received three ObTiMA scenarios, each one described different responsibilities (three different user profiles) with a list of questions to be answered online. Scenario (PG-6) was for a volunteer in the role of a data entry clerk, scenario (PG-7) was in the role of a data manager and scenario (PG-8) was in the role of a computer scientist.

The ObTiMA's web page provides a short introduction to the tool. In general, the task should only give the user a possible impression of how to work with the tool; it should not be planned as a usability test with the participatory observation of a usability engineer. On the other side the users need more extended introduction about ObTiMA to proceed effectively and efficiently.

In all these tests users succeeded in several parts and encountered various problems as well which we describe in what follows. All the occurred problems, deficiencies and violations show the designers where time and effort are needed to improve the software.

To evaluate these tests without a recording is very difficult. We should repeat at a later time these tests with a better introduction of the functionality of the tool to get more information about the usage problems and it could be important to get more experience when more than one user is working with the tool.

ObTiMA Scenario (PG-6) - The user in the role of a data entry clerk:

- Login to web page https://obtima.org/demo/summer-school
  Access is provided by a developer
- Select the trial SSPG6
- Have a look at the patients already registered into the trial
- Insert a new patient (using fictional data as reported in file CRF_UKS.pdf
  - o register the patient into the trial
  - o fill in the CRF for metastases
  - o for surgery
  - o for the surgeon

User problems with recommendations

Problem	Recommendation
The user couldn't see name and given name when viewing patients.	The name of the patient has to be available and visible for the user who enrolled it. It should be anonymized only for users who are not responsible for seeing this data. Users shall not lose time in searching for their own entered patients because they see only the anonymized identifier of the patient.
The user clicked on "Manage Patients" and only saw a list of pseudonyms, so there was no way to find the patient he just created. This is annoying if there are lots of patients and he accidently clicked on "Manage Patients".	

The user has problems by filling in a CRF for a patient. He needs more explanation when filling a CRF for a patient.	An explanation for filling in a CRF for a patient should be available.
The user doesn't know what "increase validation" means.	
When filling a form field a red sign is appearing. For the user it is a negative associated sign and it appears next to the field the user filled in. So, the reaction of the user was that there is an error with the filling of the form field.	A legend to explain the different colours should be available. See Recommendations concerning the dialogue principles (ISO 9241-110) p.44
This problem was already described in earlier usability tests	
The procedure to create a new patient and fill in the CRF for metastases was easy. But the procedure for surgery and surgeon was not clear. The case could be on one side the wrong description and on the other side the missing CRFs.	CRFs provided in ObTiMA for a patient should be equal to paper based CRFs prepared for a test person.

All these deficiencies were documented in the written use scenarios MAT1 – MAT3. The member of p-medicine has similar or same problems.

ObTiMA Scenario (PG-7) - The user in the role of a data manager:

- Login to web page https://obtima.org/demo/summer-school
  Access is provided by a developer
- Select the trial SSPG6.
- Have a look at the patients already registered into the trial.
- Could you see the personnel information like name and surname?
- What CRFs are associated to the first patient?
- Report any question you have as to be sent for the attention of the personnel at the investigating centers.

User problems with recommendations

Problem	Recommendation
For the test person it was not clear how to find the existing patient again after anonymization. The user's comment was: "More than one user is connected to this CRF, so there is no way to find again the patient." A requirement should be that	The user who entered the patient should be able to find the patient efficiently without searching under the anonymizations
The test person saw four patients in the list, the first name is visible only when selecting. It was Maria Swartz and she has only one	

CRF "SIOP2001-Metastatectomy (F3 D). When the user is allowed to see all patients, why could he not see it in the patient list?	
A difficulty for the test person was to create a new treatment plan. The test person didn't know, that for this test the treatment plan was not available for using.	A correct test scenario without creating of a patient's treatment plan should be provided for the test person.

ObTiMA Scenario (PG-8) - The user in the role of a computer scientist:

- Login to web page https://obtima.org/demo/summer-school
  Access is provided by a developer
- Select the trial SSPG6.
- Have a look at the CRFs already assigned to the trial.
- You are in the charge of adding a new CRF that can be found into the repository:
  - access the CRF repository
  - you need to select the Adverse Event Form
  - insert the needed constraints and extra queries to make sure that all the information will be entered correctly and the user will be informed when trying to introduce information that are incompatible or not in the requested format.

User problems with recommendations

Problem	Recommendation
The user clicked on the Adverse Event Form and nothing happened.	This behaviour should be approved in ObTiMA
The selected CRF was added to a list called "Selected CRFs" at the bottom of the screen. The screen was too small to get feedback from the system.	The ObTiMA interface should be adapted to smaller screens.
The test person doesn't know what the expression "needed constraints" means. It is not intuitive for him.	An explanation for using constraints in ObTiMA should be available.
When the user tried to add a constraint to the first question "Adverse Event (diagnosis (if known) or signs/symptoms)" a text field, a window telling the user that he can add a constraint by hitting "add constraint". This is annoying, because the user already clicked "add constraint" in the CRF editor.	
Comment from the user: "If there are	

"question constraints" available, this makes sense, but if there is an empty page in between, it feels clumsy."	
More important, if the user hit "add constraints" again in this scenario, he was prompted to select a constraint for "INPUT_TEXTAREA", but this list is empty.	
The user hit "Save" and ObTiMA crashed.	This behaviour should be approved in ObTiMA

A comment from another user was that adding a new CRF was an easy step by step procedure

Conclusion of the usability tests without the usability engineer

Important for these tests was the opportunity for external users to get a first impression how to work with the tool ObTiMA. For not acquainted users it would be essentially necessary to provide them with a short introduction of the main functionality of ObTiMA because of its complexity.

Real usability tests should be participatory observed by a usability engineer both to get more detailed information about the usage problems as to achieve a better evaluation of the results. It was a first experience and showed all of us to continue with real usability tests.

6   Evaluation of the ALGA Questionnaire

The evaluation and validation activities for the ALGA Questionnaire started with an item pool Generation, using questions selected from a number of validated psychological questionnaires. The initial draft of the ALGA questionnaire contained 46 items related to four themes of physical and psychological health status plus an initial set of 20 demographic questions and a final set of 6 questions regarding the difficulties eventually encountered by the subjects in filling out the questionnaire.

The investigation of norm scores from the general population facilitates the specification of cut-off scores necessary for referral.

For this first aim, the questionnaire has been administered to 778 healthy control subjects (who have never received a diagnosis of cancer).

Furthermore, the comparison of scores from healthy control subjects in the general population and cancer patients provides information about the discriminant power of the instrument concerning the detection of differences in scores of clinical and non-clinical individuals.

For this second aim 45 women operated for newly diagnosed primary breast cancer.

Healthy control subjects filled the questionnaire online, while patients answered the questions using an i-pad while waiting for the first appointment with the medical oncologist after the operation.

The total sampling population for factor analysis was (n = 823). The exploratory factor analysis (EFA) was analyzed by the principle component extraction method with varimax rotation to examine the relationships among variables without determining a particular hypothetical model. The items whose loading value was 0.45 or over were kept. Internal consistency was assessed by calculating the Cronbach's coefficient.

RESULTS

Demographic information

Healthy control subjects: The age mode was 40-49 years (23.5%). Sixty-five percent were female and 24.7% has undergone a psychological treatment in their lives. Most of the subjects were married (49.7%) and had diplomas (99%).

Patients: The age mode for patients was 40-49 years (35.6%). All of them were female, 17.8% has undergone a psychological treatment in their lives. Most of the subjects were married (75.6%) and had diplomas (73.3%).

Principal components

Factor analysis individuated the following components as relevant in explaining the personal profile: "physical well being (F1)", "body image (F2)", "sexual life (F3)", "self efficacy (F4)", "anxiety (F5)", "rumination (F6)", "cognitive closure (F7)" and "cognitive functioning (F8)" being assigned to the set of variables with high loadings on the first 8 factors (Fig. 6 and 7).
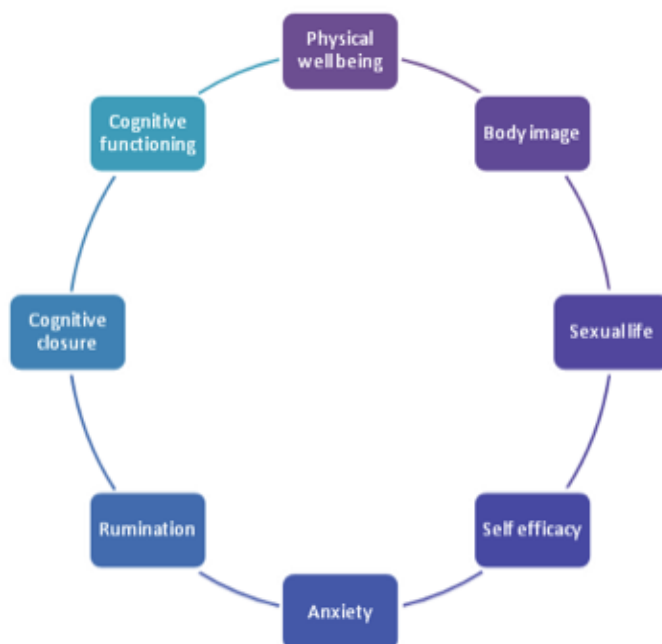
Figure 6: The 8 factors composing the ALGA questionnaire and that explain the personal profile
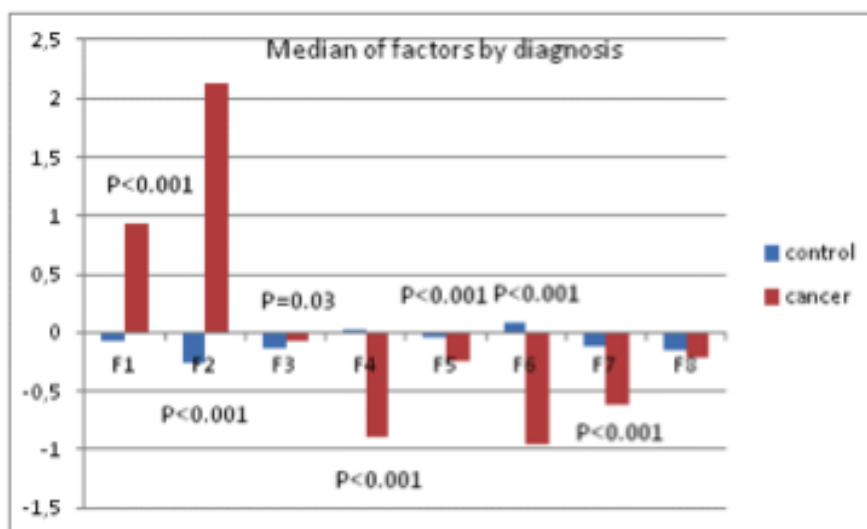
Figure 7: Median values of factors and *p*-values from multivariate ANOVA model

Finally, results showed a significant effect of the disease status on the individuated factors: Multivariate ANOVA models including socio-demographic variables (age, parity, education, marital status) and psychological treatment showed significant differences between patients and control, demonstrating that ALGA is a good instrument to be used in order to highlight those areas (factors) in which cancer patients differentiated and to which the physicians should pay attention.

7 UCL p-medicine usability workshop

During the usability workshop at UCL in London, new user needs and requirements were discovered.

Overviews of the main p-medicine tools were presented, we therefore list here the user needs in a toll-related way:

1. Portal
2. Biobank access
3. Workbench
4. Data push
5. User registration
6. Pseudonymization
7. ObTiMA
8. Patient empowerment
9. Ontology annotator
10. Ontology aggregator
11. Data mining

The aim of this meeting is to show the p-medicine end users some of the tools we are developing, to make sure they understand them, and for them to suggest changes and improvements.

Portal

The p-medicine portal provides clinicians, patients and researchers a platform to collaborate, share data and expertise, and use tools and services to improve personalized treatments of patients. The technical solution for the p-medicine portal is based on the Liferay Enterprise Portal Framework which fulfills the complex user requirements for the p-medicine portal.

The p-medicine portal is a web application providing comfortable access for the p-medicine users to the tools and services integrated into the p-medicine environment. The portal has been developed following the requirements of the user groups and their roles for the p-medicine portal.

The current version of the portal contains the initial functionality of the p-medicine security framework, Data Mining tools, p-medicine Workbench, Ontology Annotator as well as ObTiMA.

A demo server is available under https://pmedportal.ibmt.fraunhofer.de

The community based architecture of the portal has been demonstrated as well as how users can access the p-medicine tools and services integrated in the portal.

The p-medicine portal users will not need to sign in to every application separately. With a Single Sign-On functionality the login procedures of different applications is centralized in one system, accessible with only one password. After entering the right password, only the applications that the user is authorized to will become available. Within each application the internal permission system is used.

Users of the p-medicine portal can be grouped together to communities providing a powerful mechanism for the portal administrator to configure portal resources and security in a consistent and robust manner. Communities are collections of users that have a common interest (e.g. different portal users are members of a community called SIOP Community that has a common interest in the nephroblastoma diseases). Membership in communities gives

users access to the pages in the communities of which they're members. Each community can have a specific layout and an own set of available pages containing different p-medicine tools.

Discussion

It has been agreed that the interface for the forum should be renewed because as it is very technical and difficult to use.

Import of documents and media should be discussed also with the legal partner to avoid copy right problems.

Biobank access

A Biobank Access Framework called p-BioSPRE has been developed within the p-medicine platform to enable and simplify access to existing biobanks but also to offer own biomaterial collections to research communities and manage biobank specimens over the ObTiMA Trial Biomaterial Manager.

The main component of the framework is p-BioSPRE, the p-medicine Biomaterial Search and Project Request Engine, which is a metabiobank to share biomaterial for research purposes. Furthermore, the framework comprises the p-Biobank Wrappers, which are tools to support biobank owners to offer their biomaterial and related data, (regardless which biobank information system they are running on-site), in p-BioSPRE and manage associated requests. In order to enable users of the p-medicine trial management system ObTiMA to integrate biomaterial data in clinical trials and offer it in p-BioSPRE a Trial Biomaterial Manager is provided.

By using the interface a trial chairman or biobank owner can export the specimen data of a trial to provide it in p-BioSPRE. For this purpose the data firstly needs to be loaded into his p-Biobank Wrapper installation. To this end the user can select a trial and all specimen data stored in the trial and the data is exported pseudonymized into a file in the CDISC ODM format. The user can upload the exported file into his p-Biobank Wrapper installation.

Discussion

The interface and functionalities were much appreciated by the users. A critic was raised in order to highlight the fact that the biobank update depends on the local associated biobanks.

Workbench

The p-medicine workbench is the end-user application that provides access to various p-medicine tools for the clinicians to use and invoke. This tool is basically an "application store" where different applications, tools, and services are categorized and indexed accompanied with information about their functionality, operation, development status, accessibility, etc.

The primary users of this application are the clinicians but of course it would be also useful to other user groups such as bioinformaticians etc.

The Workbench aims to boost the communication and collaboration of researchers among Europe for the machine-assisted sharing of expertise. It is a web based integrated and collaborative environment, where all the relevant to P-medicine tools can be discovered, published and annotated semantically and with proper metadata. The Workbench is part of the p-medicine's portal (it is a Portlet).

Discussion

Critics were raised on the usability of the interface, the clinicians found the interface too complicated to use by their own.

Data push

The Data Upload tool is a desktop (java) graphical application for "pushing" data stored in files through the security infrastructure to their final destination, which is the p-medicine data warehouse. It strives to be a user-friendly tool hiding a lot of complexity behind its simple user interface. **Its main functionality is to allow a user to load a file from the local file system, and upload it to the security infrastructure for (pseudo) anonymization**. The security infrastructure takes care then to go over multiple anonymization steps and finally submits the data to the data warehouse. The following Figure shows the whole data upload process and the role of the Upload Tool in this process:
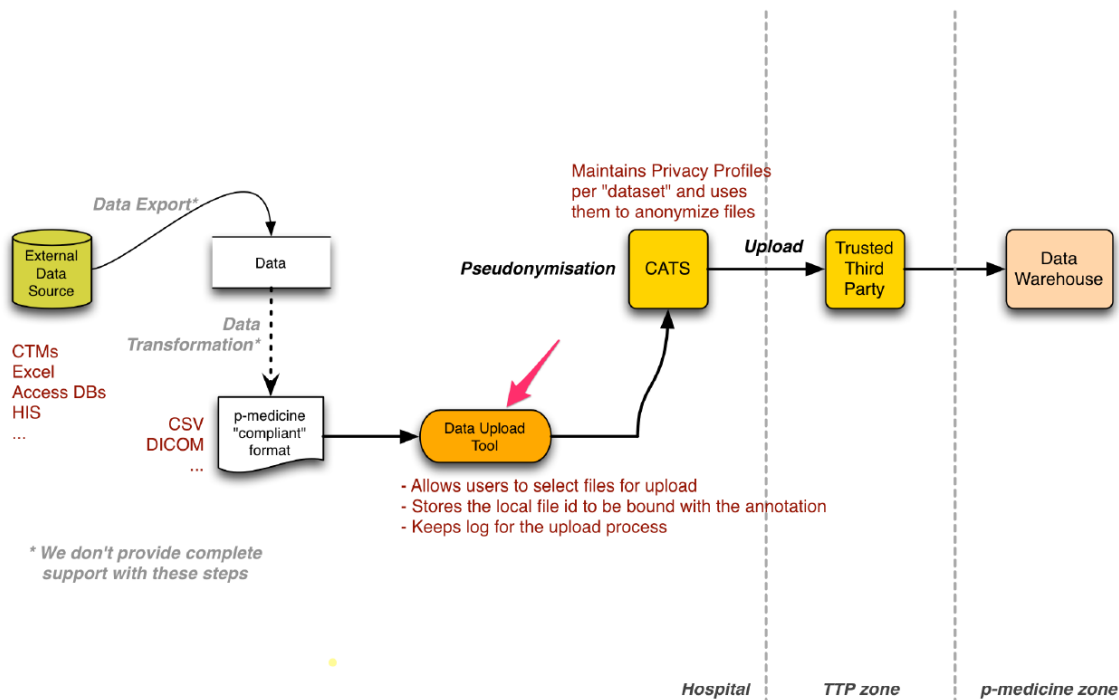
Discussion

The tool has been received very well by users, the developers will move forward with the integration of the tool with the other components in order to create for the end user an interface were he can upload the files on the data warehouse in a complete transparent way.

User registration

Public registration

If a user wants access to the p-medicine framework, the user must have a valid p-medicine account on the Security Framework. The user can surf to the registration URL of the IDM. There he/she will provide name, email, address, phone, etc. Afterwards an administrator is notified that a new user wants to receive an account. The administrator will review the pending registration request and accept or deny the user. In case the user is accepted to the p-medicine framework, the user will receive an activation mail containing an activation link. Clicking on this link will open a browser window in which the user can submit a username and password. Now the user can authenticate to different service providers within the framework.

Private registration

An identity administrator can decide to register a user. In this case the user receives the same activation mail as in the public registration scenario.

Discussion

The system is very robust, the only discussion raised in order to determine a better evaluation of the new user system than the administrator evaluation, still under discussion.

Pseudonymization

The pseudonymisation tool (CATS) is a background service responsible for the deidentification of the data files before being stored in the data warehouse. The tool will take any medical file as input and transform it into a pseudonymised file by clearing all identifying data and including a pseudonym instead.

Although CATS will mainly run as a service, CATS has a web frontend in which a user may monitor the state of a file. During processing the file goes through different phases and states. If the processing of a file should fail, a user can reprocess the file.

CATS is configured by means of a set of "Privacy Profiles". These profiles contain transformation rules that CATS needs to process clinical data. Rules can be as simple as "clear the first name" or "encrypt the address", or more complex like "replace all identifying data by a random pseudonym" or "replace the treatment dates by a date relative to the date of birth".

To create (and test) the privacy profiles we use a tool named CAT. This is a client side graphical application allowing any user to create, modify and store privacy profiles.
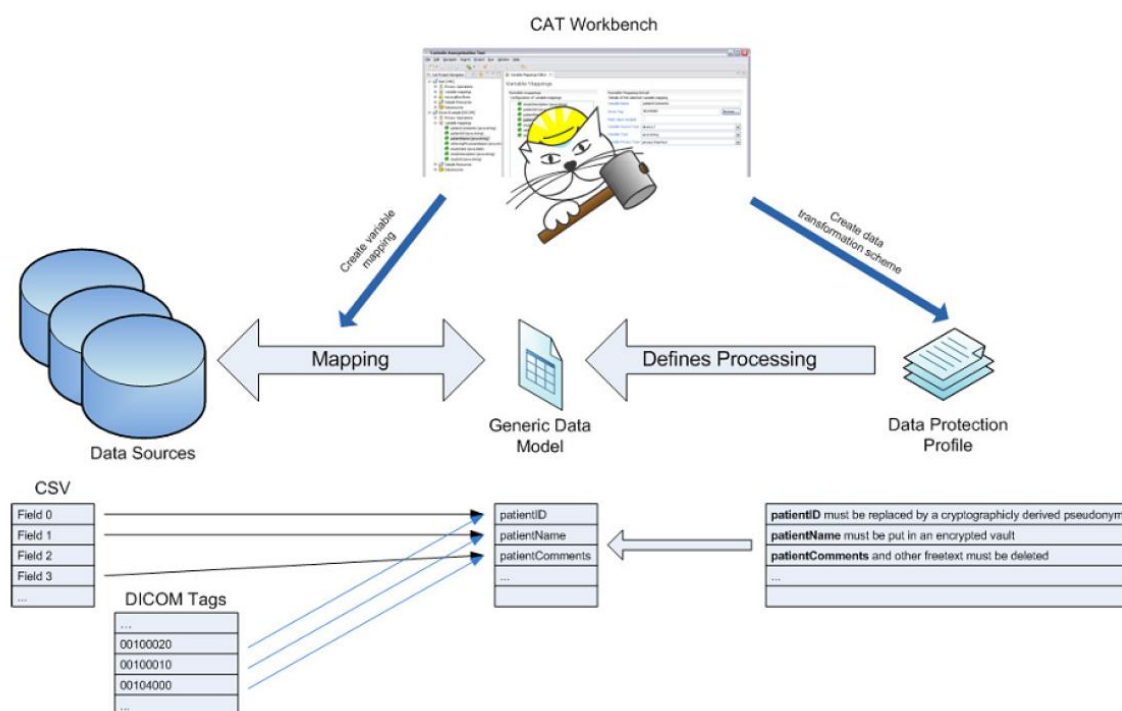
Discussion

The tool has been received very well by users, the developers will move forward with the integration of the tool with the other components in order to create for the end-user an interface were he can upload the files on the data warehouse in a complete transparent way.

ObTiMA

ObTiMA has been already extensively described here and in previous deliverables, therefore we omit the description from this paragraph.

Discussion

The users were very pleased to see such an appealing tool. No critics have been raised.

Patient empowerment

Alga questionnaire has been already described here and in other dedicated deliverables.

The ALGA questionnaire will be one of the tools listed on the p-medicine portal, it will be thus integrated.

Discussion

The users were very positive toward the developed interface and visualization in order to summarize the results.

Ontology annotator

The OA has been already extensively described here and in previous deliverables, therefore we omit the description from this paragraph.

Discussion

The users found the tool difficult to be used by a clinician, although the main user will be the data manager, with many technicalities that will discourage the health practitioners from using it regularly. It has been decided, in order to enhance the tool usability to associate a clinician to the OA developer, in the scenario of mapping the dataset in input by the data manager and clinician together.

Ontology aggregator

The Ontology Aggregator is a tool associated to the Ontology Annotator in order to give the user the possibility to import an ontology term in case the latter is not already available in the HDOT.

Discussion

The participants agreed in expanding the Ontology Aggregator to the set of terms that are envisaged as to be needed for the p-medicine pilot studies. The developers will continue working on the integration with the Ontology Annotator.

Data mining

Several tools were introduced to the audience in order to receive a user's feedback. Notably, the text mining tool will be used in collaboration with ecancer to extract knowledge from patients' forums.

8   Appendix A

Task of the data manager

Weiße und gelbe Seiten direkt an die Studienleitung schicken - grüne Seiten für die Patientenakte.

NEPHROBLASTOM – THERAPIEOPTIMIERUNGSSTUDIE SIOP 2001/GPOH – OPERATIONSBOGEN – METASTASEN

Studienleitung: Prof. Dr. N. Graf, Universitätsklinik für Kinder- und Jugendmedizin, 66421 Homburg/Saar
Tel: 06841 16-28047, 28397, -28000   Fax: 06841 16-28302   Email : norbert.graf@uniklinik-saarland.de
in Zusammenarbeit mit dem Deutschen Kinderkrebsregister am IMBEI, 55101 Mainz
Tel.: 06131/17-3227   Fax: 06131/17-4462

Name/Aufnahmenummer: *Peter Maier – Landruth*
Pat.-Nr. /-Zahl in der Studie: 1 2 9 5 N 0 1 7 1
Klinik
Identifikationszahl vom IMBEI: 0 9 0 3 0 5 1 2 3 4 5 44
Geb.-Datum

GPOH-PID: A B C 1 2 3 D E Klinik

Operationsdatum: 1 9 0 6 1 2		**Chirurg:** Dr. Steven Schmidt	

Operateur:
☐ Kinderchirurg
☒ Thoraxchirurg
☐ Allgemeinchirurg
☐ Urologe

Klinik: UKS
PLZ, Ort: 66424 Homburg

Indikation: kurativ ☒   palliativ ☐

Präoperative Therapiemethode:
Chemotherapie   Ja ☒   Nein ☐
Radiotherapie   Ja ☐   Nein ☒
Metastasektomie   Ja ☐   Nein ☒

Lokalisation der Metastasen:
Lunge ☒   Knochen ☐   ZNS ☐
Leber ☐   Weichteile ☐   sonst ☐

Genauer Ort: _____

Auftreten der Metastasen zum Primärtumor: synchron ☒   metachron ⊠ West februh

Seite: rechts ☒   links ☐

Metastasektomie erfolgte in:
Lunge ☒   Knochen ☐   ZNS ☐
Leber ☐   Weichteile ☐   sonst ☐

Verwachsungen: Nein ☒   Ja ☐

Entfernung:
komplett ☒   Anzahl 0 2
inkomplett ☐   Anzahl ___
nur biopsiert ☐

Genaue Angaben für Lunge und Leber:
weite Resektion ☒   Segmentresektion ☐
Lobektomie ☐   Pneumonektomie ☐
Transplantation ☐

Weitere Therapie:
keine ☐   Chemotherapie ☒   Radiotherapie ☐
Re-OP ☐   Stammzelltransplantation ☐

Dieser Dokumentationsbogen ist vom Chirurgen unmittelbar nach Operation auszufüllen. Der Operationsbericht ist in Kopie an die Studienleitung zu senden. Für jede Metastasenoperation (F3b) ist ein gesonderter Bogen auszufüllen.

Bemerkungen:

Stempel: **Datum:**   **Unterschrift**

Version 2.0 / August 2002

9 Appendix B

Screenshots of the first and second prototype of ObTiMA

Screenshots from the first prototype of ObTiMA: usability test M1

Legend: The yellow circle ⬤ marks the cursor position on the website

Figure B1M1: Welcome to ObTiMA

Figure B2M1: Sub items of Trials are shown
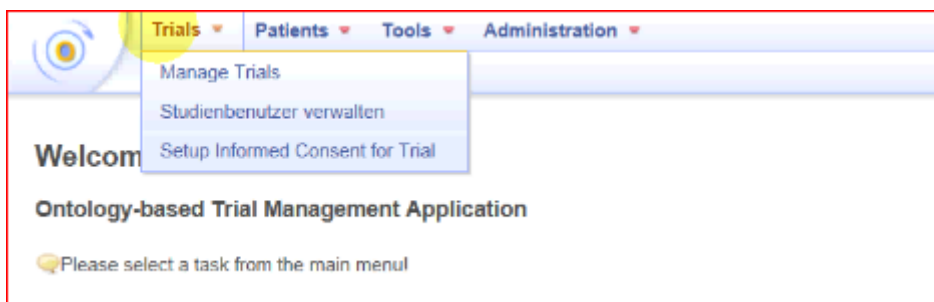
Figure B3M1: Sub item Manage Patients is shown under Patients
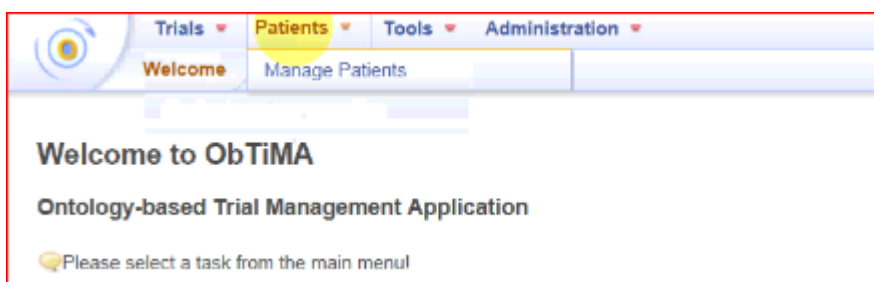
Figure B4M1: Existing Trial SIOP2001/GPOH to manage patients

Figure B5M1: List of Patients (only one) under Trial 'SIOP2001/GPOH

Figure B6M1: CRF of fictive patient

Figure B7M1: Window after Click on "CRF"

Figure B8M1: List of CRFs (only one CRF)

Figure B9M1: Preview of the "Demographics" Section of BaselineDataSIOP

Figure B10M1: Section "Operation Details" of the BaselineDataSIOP is shown

Screenshots from the first prototype of ObTiMA: usability test M2

Legend: The yellow circle ● marks the cursor position on the web page.

Figure B1M2: Welcome to ObTiMA
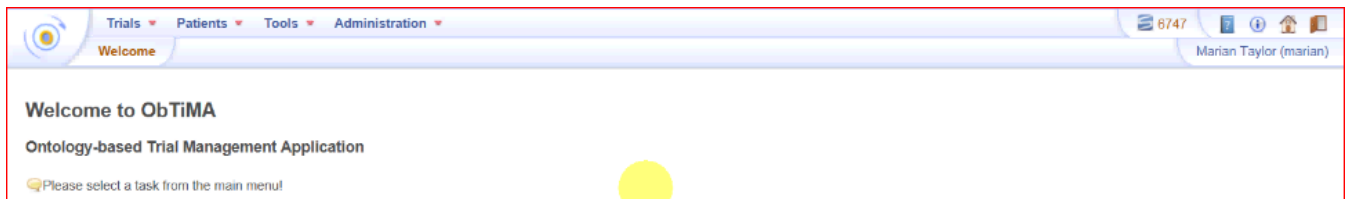
Figure B2M2: Manage Patients under a Trial

Figure B3M2: List of Patients (only one) under Trial 'SIOP2001/GPOH
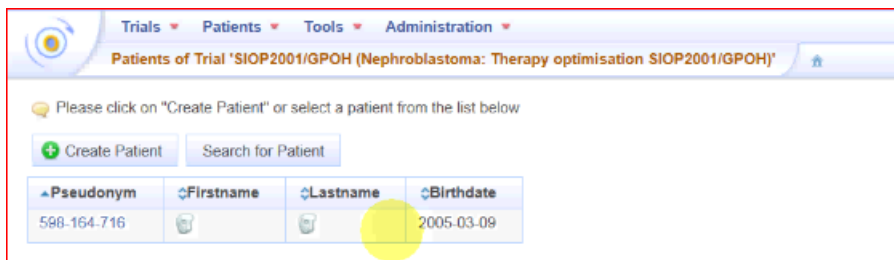
Figure B4M2: Search for Patient
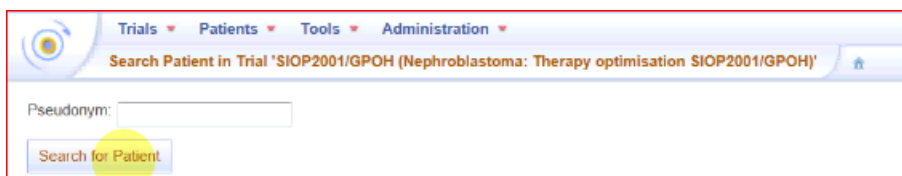
Figure B5M2: List of CRFs but no CRF exists
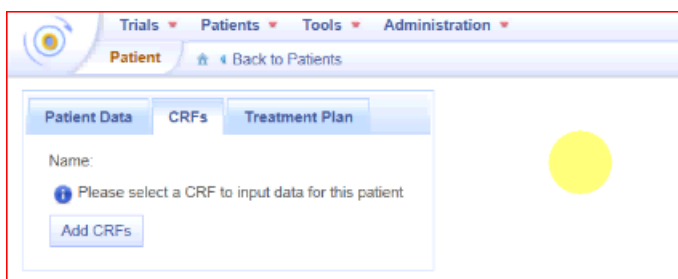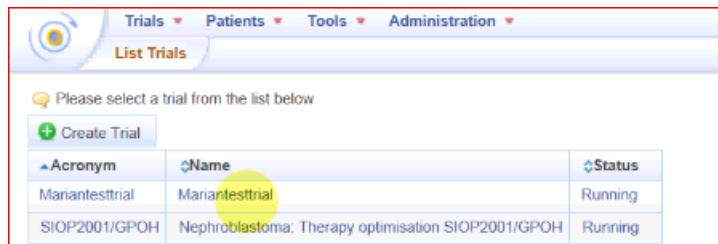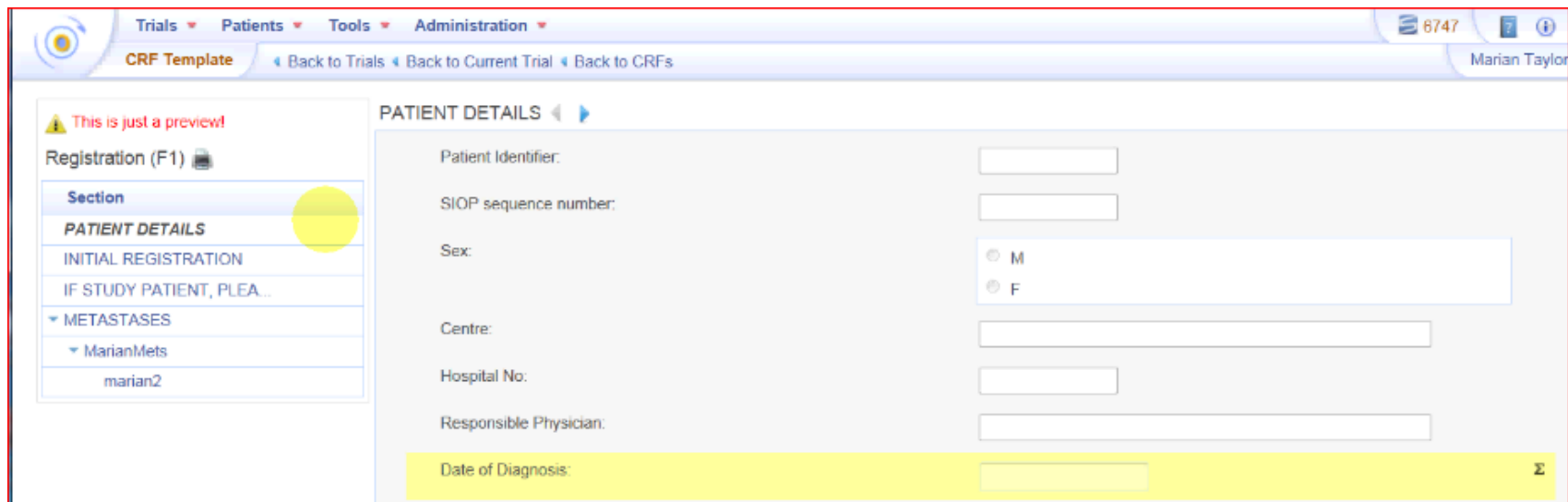
Figure B6M2: List of Trials

Figure B7M2: Preview of the CRF named Registration (F1) under the Trial Mariantesttrial
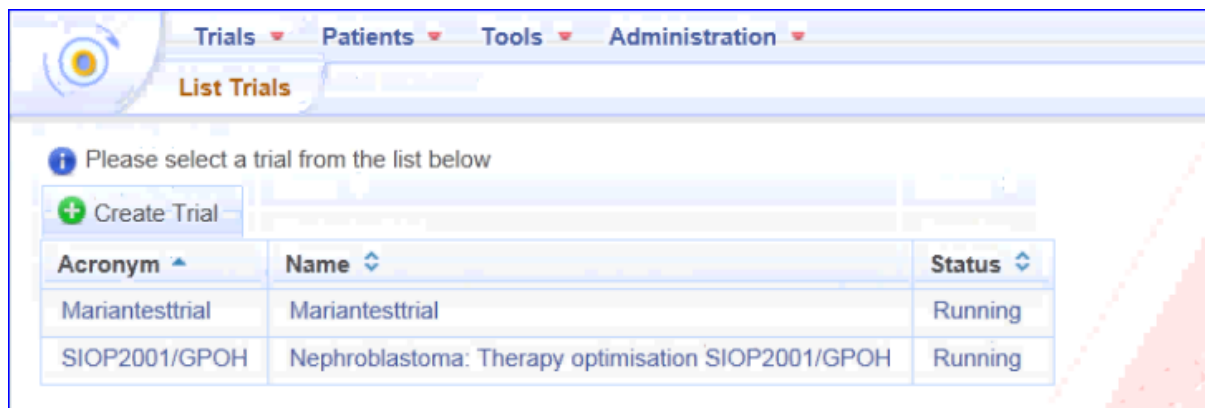
Screenshots from the second prototype of ObTiMA: usability test MAT1

Legend: The yellow circle ⬤ marks the cursor position on the website.

Figure B1MAT1: Starting with trial list

Figure B2MAT1: SIOP2001/GPOH trial data

Figure B3MAT1: Pseudonyms of patients

Figure B4MAT1: Entering patient data with a given pseudonym into Trial SIOP2001/GPOH

Figure B5MAT1: Complete patient data can be saved

Figure B6MAT1: Patient data after saving

Figure B7MAT1: User selects her current entered patient (to guess!)

Figure B8MAT1: BaselineDataSIOP has to be selected

Figure B9MAT1: Marked BaselineDataSIOP and click on "Assign Selected CRFs to Patient

Figure B10MAT1: CRF BaselineDataSIOP with red sign

Figure B11MAT1: Exception occurs when clicking on BaselineDataSIOP

Screenshots from the second prototype of ObTiMA: usability test MAT2

Legend: The yellow circle ● marks the cursor position on the website.

Figure B1MAT2: Starting with fictive patient data
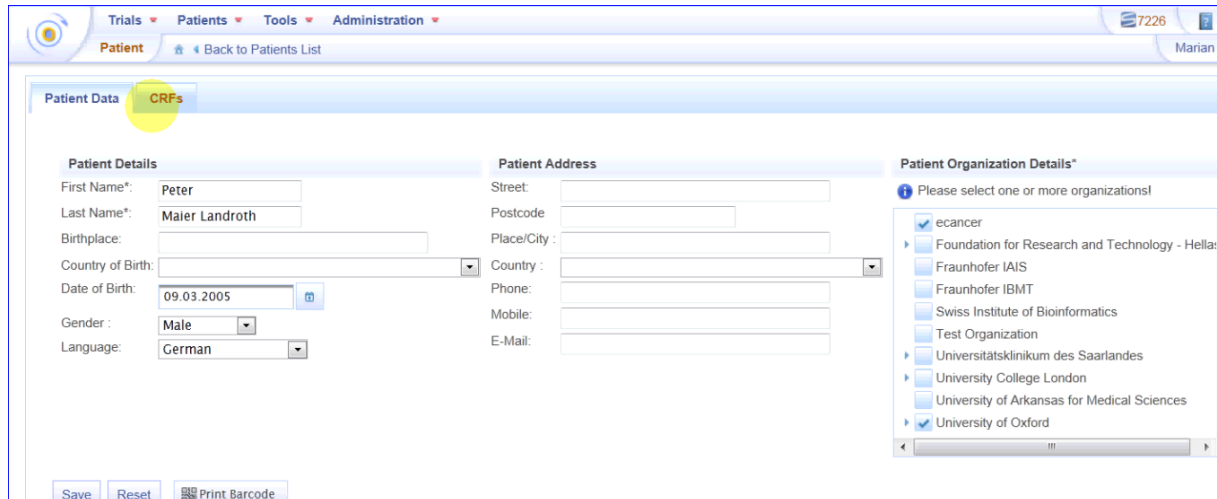
Figure B1MAT2a: Message: No records found

Figure B2MAT2: Patients list is visible via pseudonyms

Figure B3MAT2: Message: No rights to see or add CRFs

Figure B4MAT2: Content of the CRF section demographics

Figure B5MAT2: Section for operation details

Figure B6MAT2: Two running trials; selection on the SIOP2001 trial

Figure B7MAT2: Two CRFs with different status

Figure B8MAT2: List of CRFs from Repository

Screenshots from the second prototype of OBTiMA: usabiltiy test MAT3

Legend: The yellow circle ● marks the cursor position on the website.

Figure B1MAT3: Login website; the user starts again to carry on trial and CRFs to the patient

Figure B2MAT3: Two running trials
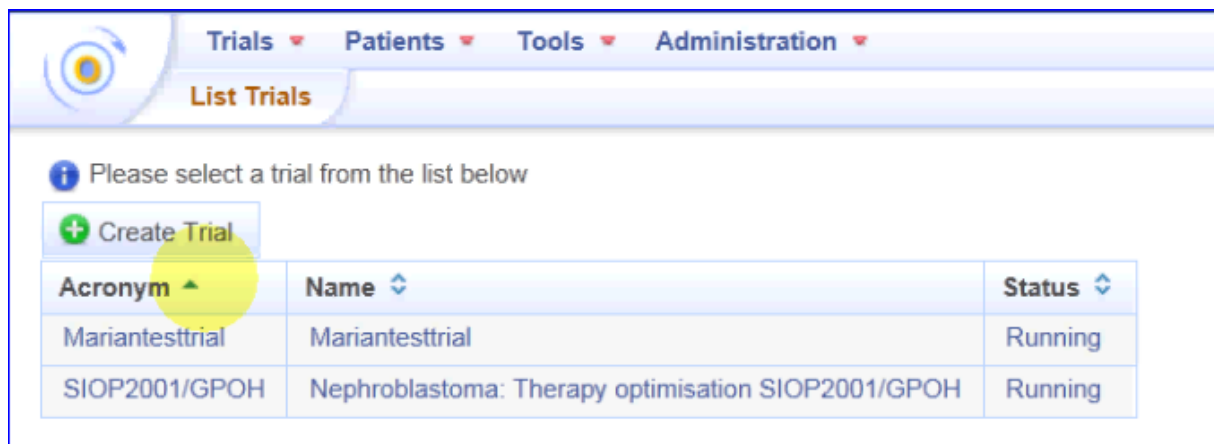
Figure B3MAT3: Available CRFs
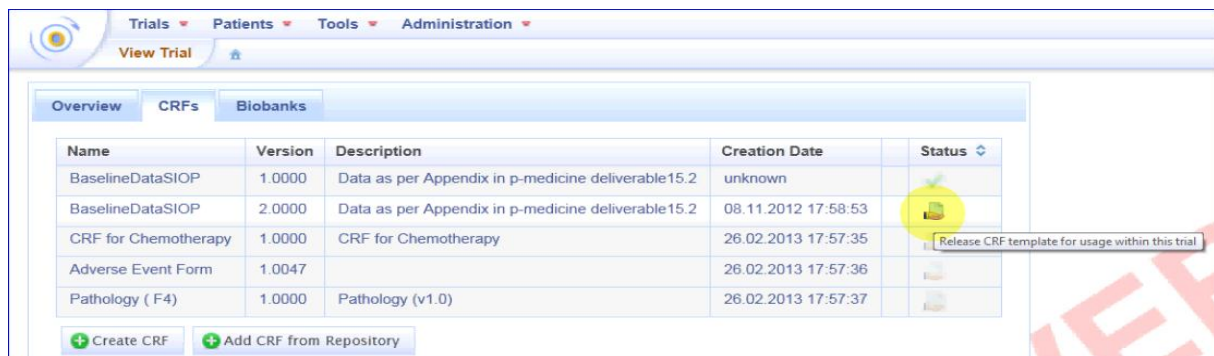
Figure B4MAT3: Pop-up window for releasing the CRF
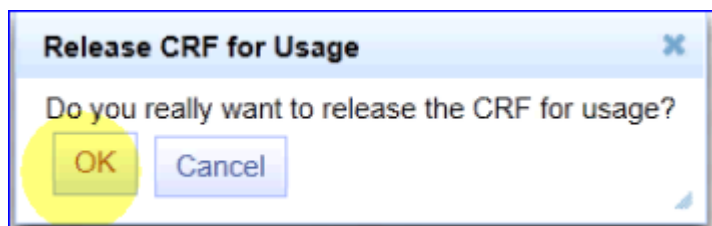
Figure B5MAT3: Released CRFs
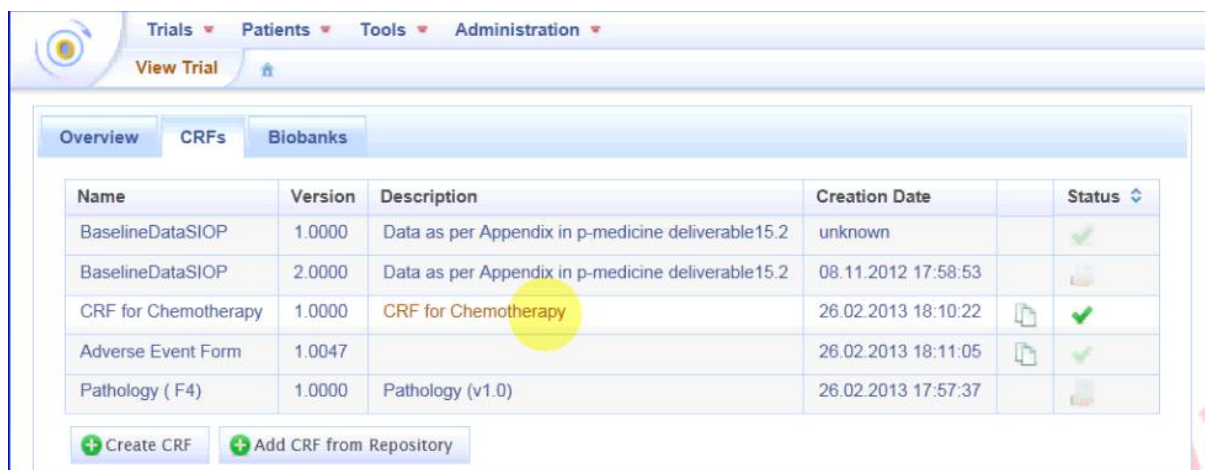
Figure B6MAT3: Information about missing question.
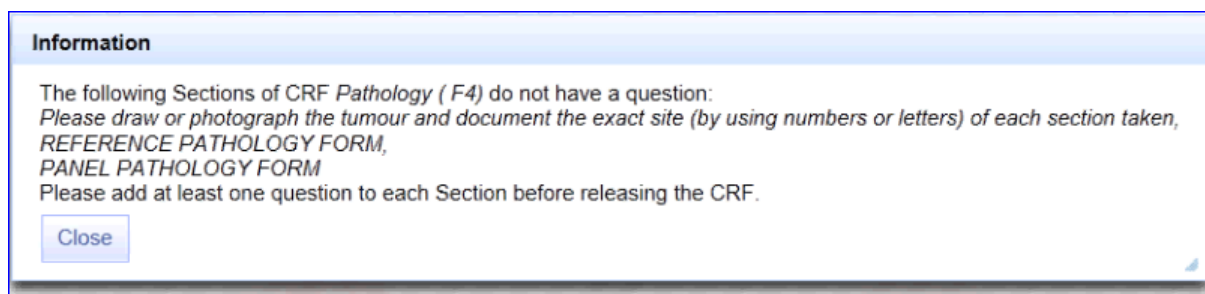
Figure B7MAT3: List of patients (pseudonyms)

Figure B8MAT3: Marked released CRFs for assigning to the patient

Figure B9MAT3: Details of CRF Chemotherapy Group

Figure B10MAT3: Not possible to enter the values for T and N classification

Figure B11MAT3: After save of the CRF for Chemotherapy

Figure B12MAT3: No records found

Figure B13MAT3: Assign Selected CRF to Patient

10 Appendix C

Screenshots of the second prototype of the OA: usability test SS13

The cursor has not been highlighted in this session.

Figure C1: Pop-up window of the OA after clicking on the '+' sign for entering the name of the project, the description and the type of the database.
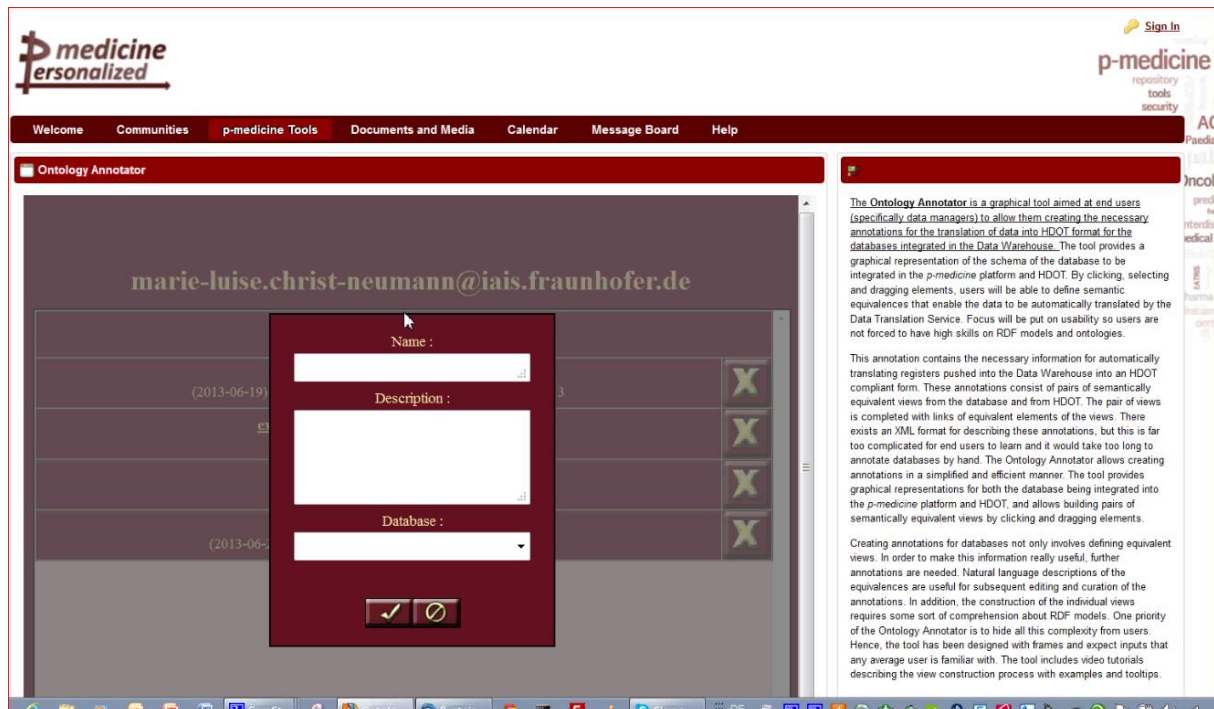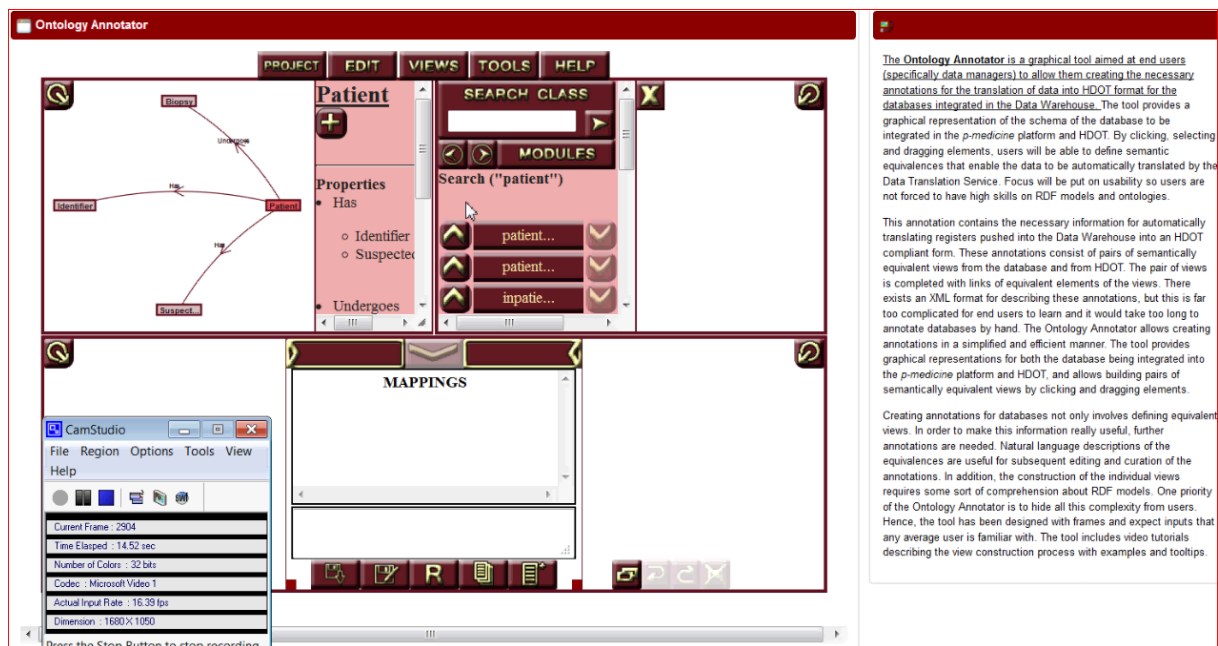

Figure C2: After reading the introduction the user continues with the task

The five visible buttons on top of the window (PROJECT EDIT VIEWS TOOLS and HELP) show the improvement in contrast to the first prototype. The user is now aware of these buttons during the whole session and this gives him a better orientation.

Figure C3: The pop-up window shows the different modules of the ontology library for selection.
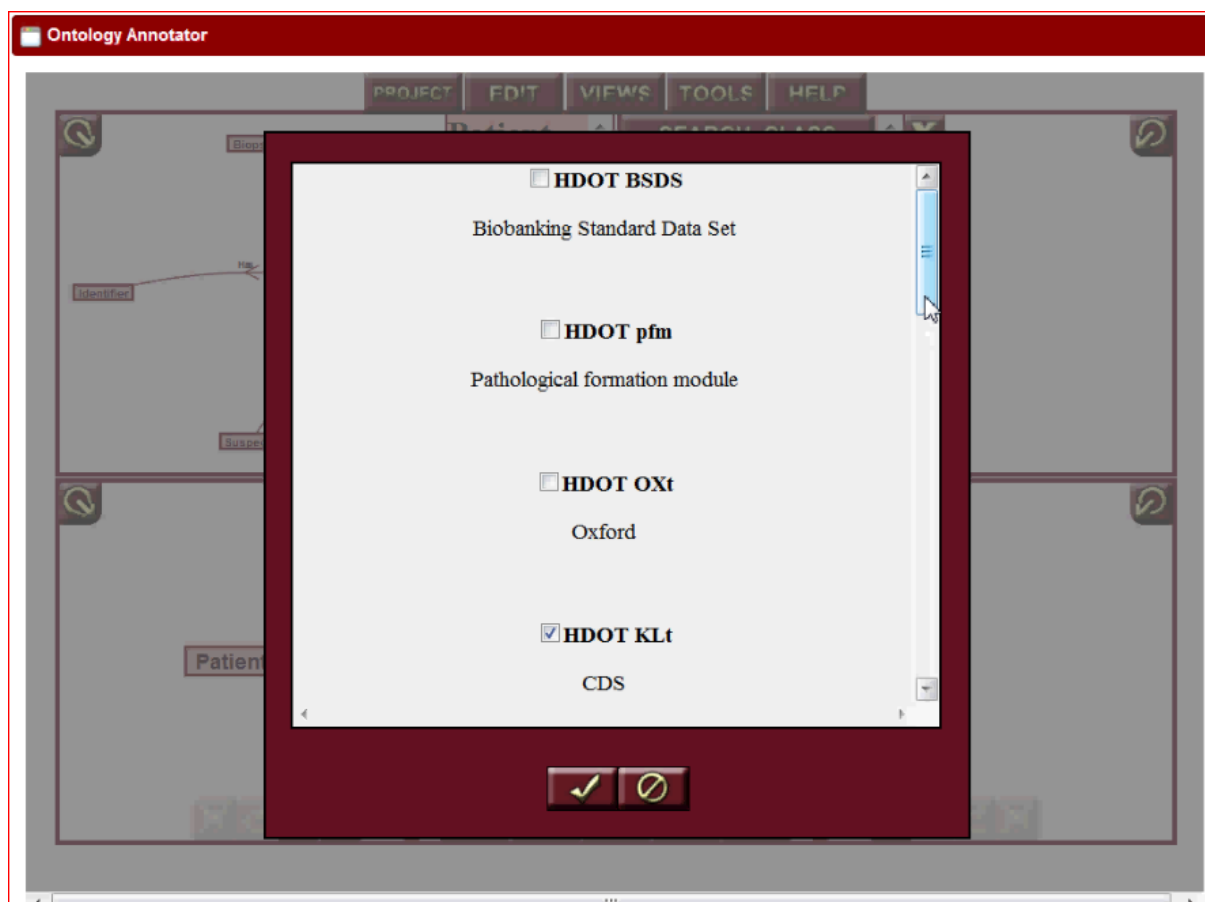

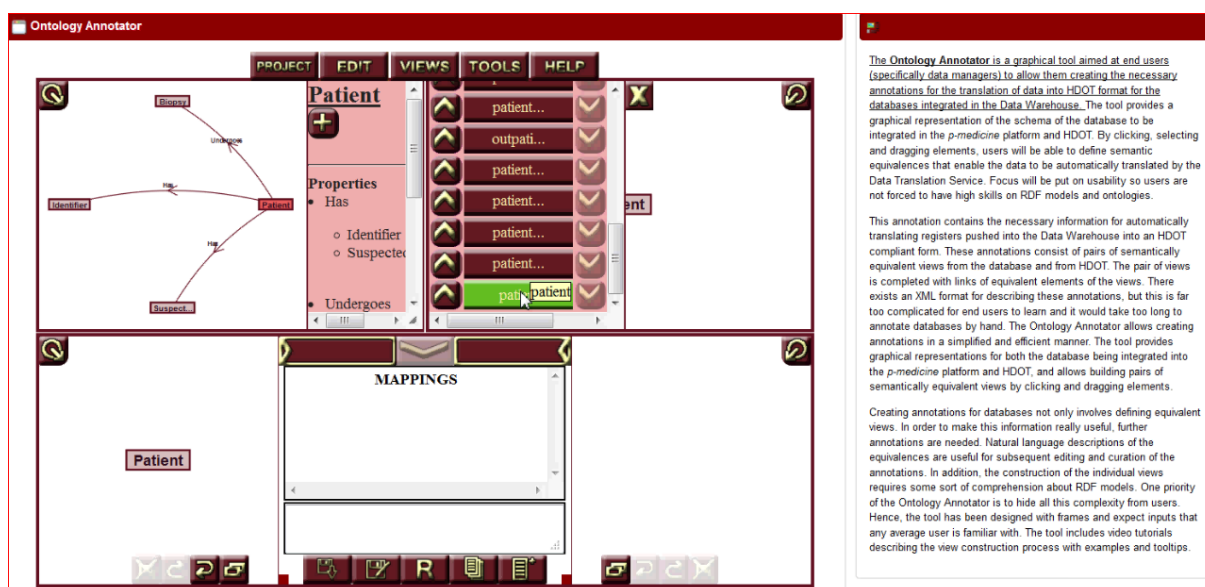Figure C4: The partly presented class "patient" in the HDOT canvas.

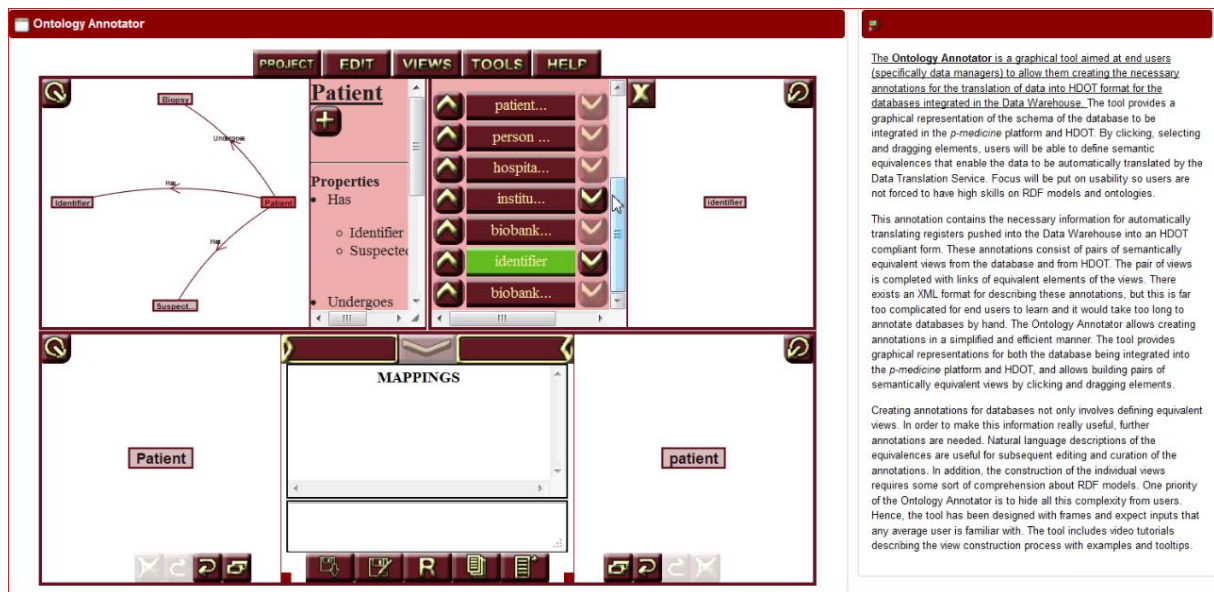Figure C5: The visible class "identifier" but class "patient" is hidden.

Figure C6: After entering the class name "biopsy" and clicking on the "search" button.
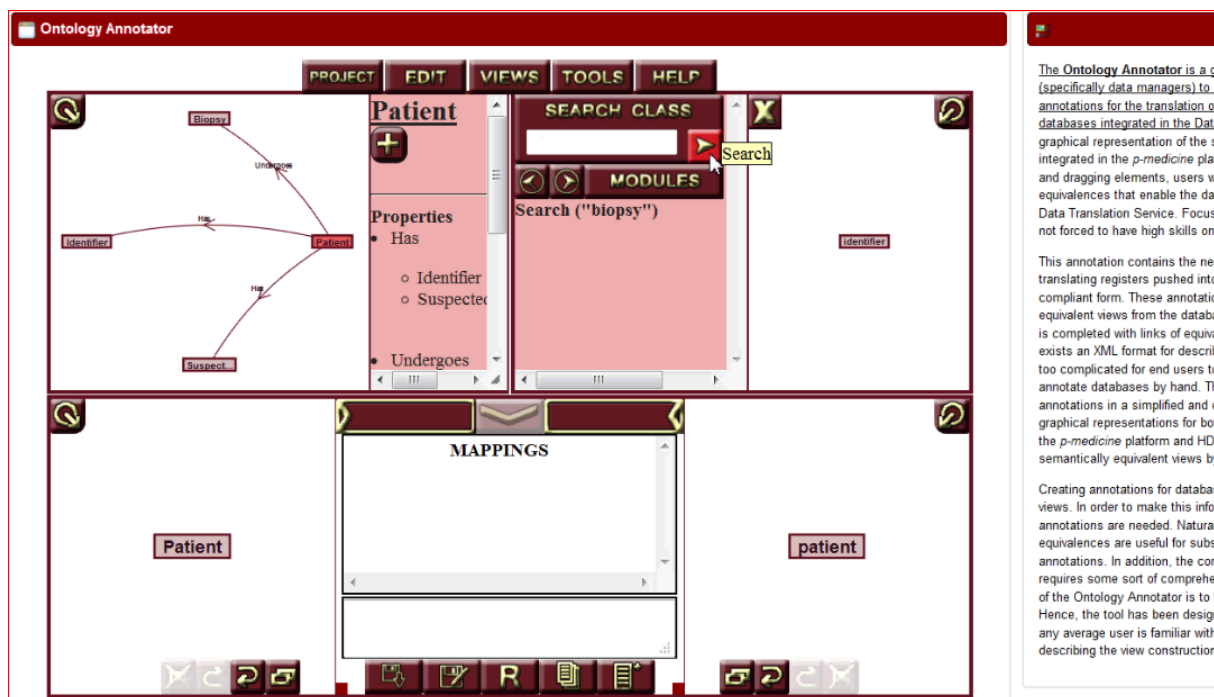
Figure C7: Searching for the class "suspect tissue" but the system has not finished searching for "biopsy" (s. line "Search ("biopsy")".
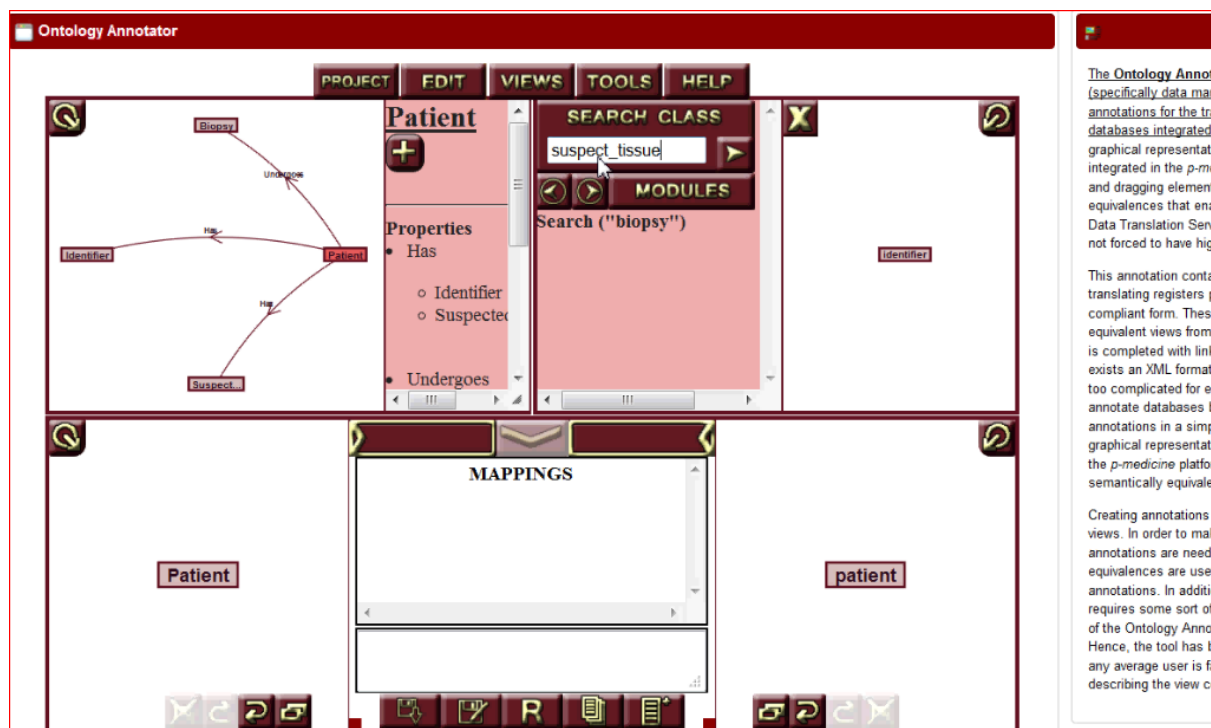

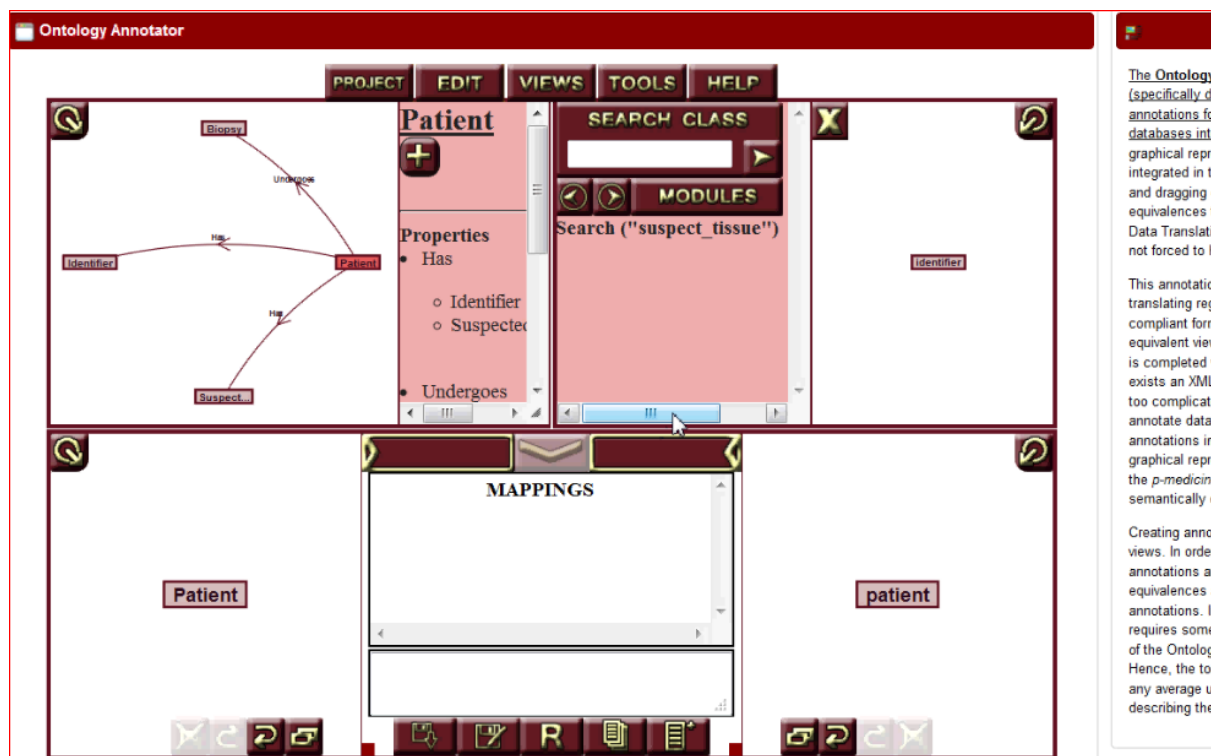Figure C8: The search action for "suspect_tissue" has not yet finished.

Figure C9: Scrolling in every direction to see the corresponding properties of class patient.
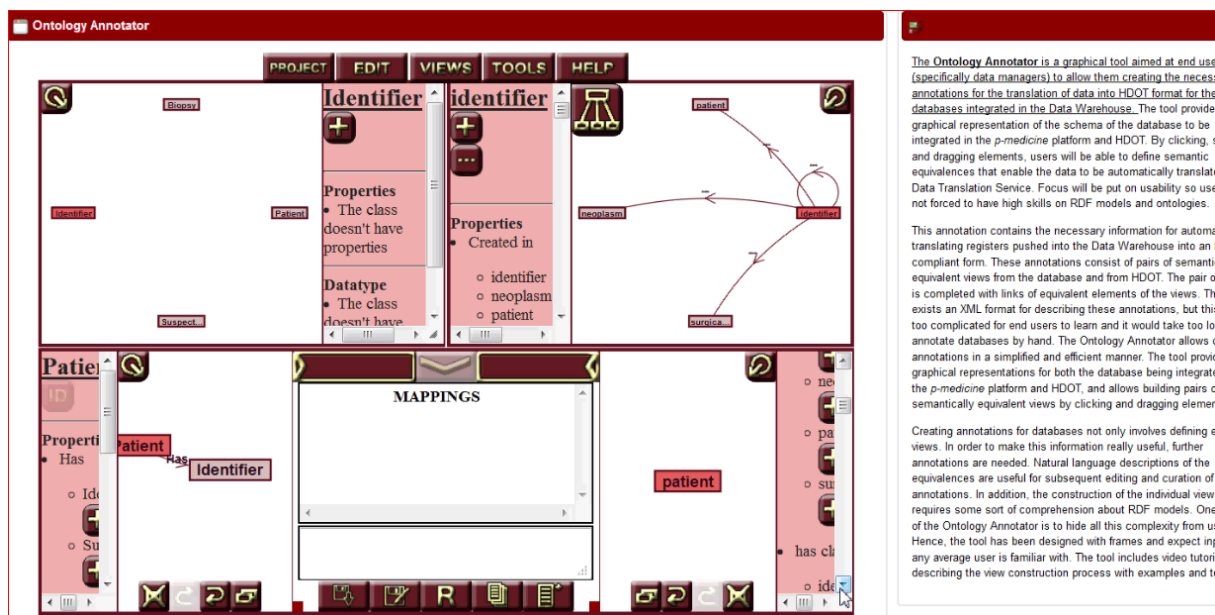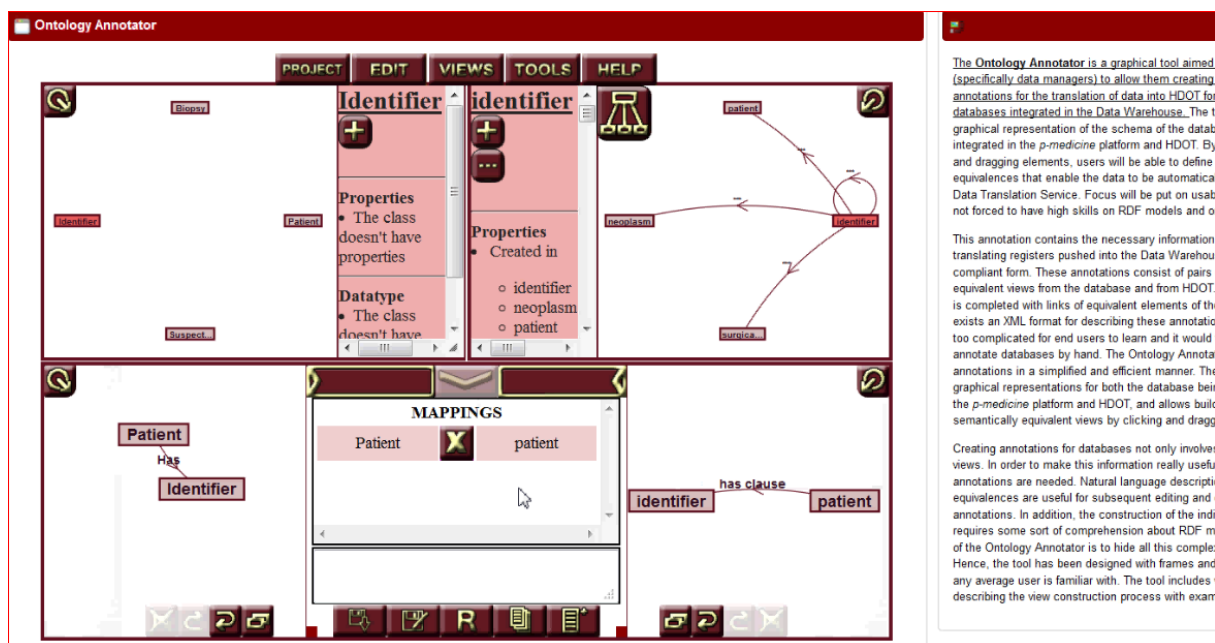

Figure C10: Shows the newly created link of class patient in the "Mappings" window which has the same meaning.

11 Appendix D

Ontology Annotator tutorial document

Introduction

The Ontology Annotator (OA) is a web-based graphical tool targeted at p-medicine end-users and that allows manually defining annotations for external databases. These annotations will allow the subsequent integration of the annotated database into the p-medicine infrastructure.

By annotation, what we mean is the alignment of the database elements with the ontology libraries used within p-medicine. The idea is that the users employ the OA to establish relations between semantically equivalent terms of the annotated database and the ontology libraries. The ontology libraries can be seen, in fact, as another database schema.

Using the Ontology Annotator

The OA is built around the concept of alignment of RDF schemas. In case you are unfamiliar with the concept of alignment and the concept of RDF, please read the next paragraphs carefully.

RDF

RDF is a language used in biomedical research (among other areas) to describe specific knowledge domains. RDF is an open standard developed by the W3C consortium (http://www.w3.org/RDF/). It was selected upon other choices due to its versatility and the ability of machines to easily process this language. RDF has become the *de facto* standard for describing any sort of resources in the web in an attempt to enhance interoperability within different domains. In the biomedical area, RDF is being used to codify data from different databases with a common language so that researchers have less trouble working with disparate databases at the same time (think it as an equivalent of using the occidental character set for writing documents instead of Cyrillic, Chinese character set and Greek character set).

What does an RDF-compliant database look like? The RDF standard is built around two principal ideas: class hierarchy and relations between classes. First, a class in RDF refers to a single specific concept (e.g. "Patient", or "Biopsy"). An RDF class models one concept in our knowledge domain. RDF classes are organized hierarchically, meaning that, each class in an RDF database can have a parent class and several children classes. This hierarchy establishes "is-a" relations between the RDF classes (e.g. we can have a "HumanBeing" class representing a person in general, and two "Patient" and "Doctor" classes as its children, representing the part of those persons that are patients and doctors, respectively). Second, RDF classes are connected to each other by means of RDF relations (e.g. the relation "treats" can join the class "Doctor" with the class "Patient, or the relation "undergoes" can join the class "Patient" with the class "Biopsy"). By specifying the set of RDF classes and its hierarchy first, and the relations linking those classes, RDF allows to formally describing a

specific area of knowledge (e.g. one RDF database can contain the necessary classes and relations to describe the area of clinical trials on cancer).

By "selecting" some classes and relations of an RDF database, we are referring to one specific idea within the whole database. This is called "creating a view". For example, within the database of clinical trials on cancer there are over a thousand classes and over a hundred relations. Among them, we can pick the classes "Patient" and "Biopsy", and the relation "undergoes" which links both of them. Therefore, we can define the view "Patient undergoes Biopsy". This view refers to the information reflecting patients that undergo a biopsy within the database.

Alignment of RDF schemas

By using the RDF standard we have all databases in the same character set, using the previous analogy. Still, each database will probably use its own language for describing the data. Where one database codifies the sex of patients with a '0' and '1', another database uses 'M' or 'F'. Where one database uses the term 'Patient' to refer to the patients of their clinical trial, another database uses 'CTPat'. Where one database one database proposes the view "Patient undergoes Biopsy" to refer to those patients that undergo a biopsy, another one uses "Patient undergo MedicalTreatment ofType BiopsyProcedure". We can regard this as the equivalent of reading different documents each in a different language (although they are all written with the occidental character set). To allow a person who wants to read all these documents without forcing her/him to learn a myriad of languages, the most obvious process is to automatically translate all the documents into a common language (e.g. English). With the disparate databases we will do the same: define annotations (dictionaries) between each database schema and a central RDF schema that acts as a common language so that the databases can be automatically translated.

The OA is the tool that allows defining these annotations. In essence, the user labour in order to annotate a database consists on defining a view on the database, an equivalent view on the central RDF schema, and then establish links between the individual classes of those views. This task is then repeated for as many views are needed to annotate in the database.

In p-medicine, we will a specific RDF schema built within the p-medicine project itself: HDOT. HDOT stands for Health Data Ontology Trunk, and provides classes and relations for describing the area of medical clinical trials.

Accessing the OA

To access the OA, you must login into the p-medicine portal (https://pmedportal.ibmt.fraunhofer.de/). Please make sure you first obtain the necessary credentials for login into this page. Once in the p-medicine portal, go to p-medicine Tools → Ontology Annotator.

The OA maintains a series of "annotation projects" for each user (or simply, projects). One project refers to the annotation of one single database. A user can work on a project throughout different sessions, and the OA will maintain the project changes between those sessions. In the first OA screen, the user is presented with the list of existing projects, and the possibility of opening one of the projects, deleting one of those projects, or creating a new one. To create a new project, the user must provide a name, a description, and select a database to annotate (from a drop-down list). The user must also specify the type of

database that she/he wants to annotate (currently, comma separated values, Access, or RDF itself are supported).

When a new project is created, or an existing one is opened, the user gets access to the main editing window, as seen below:
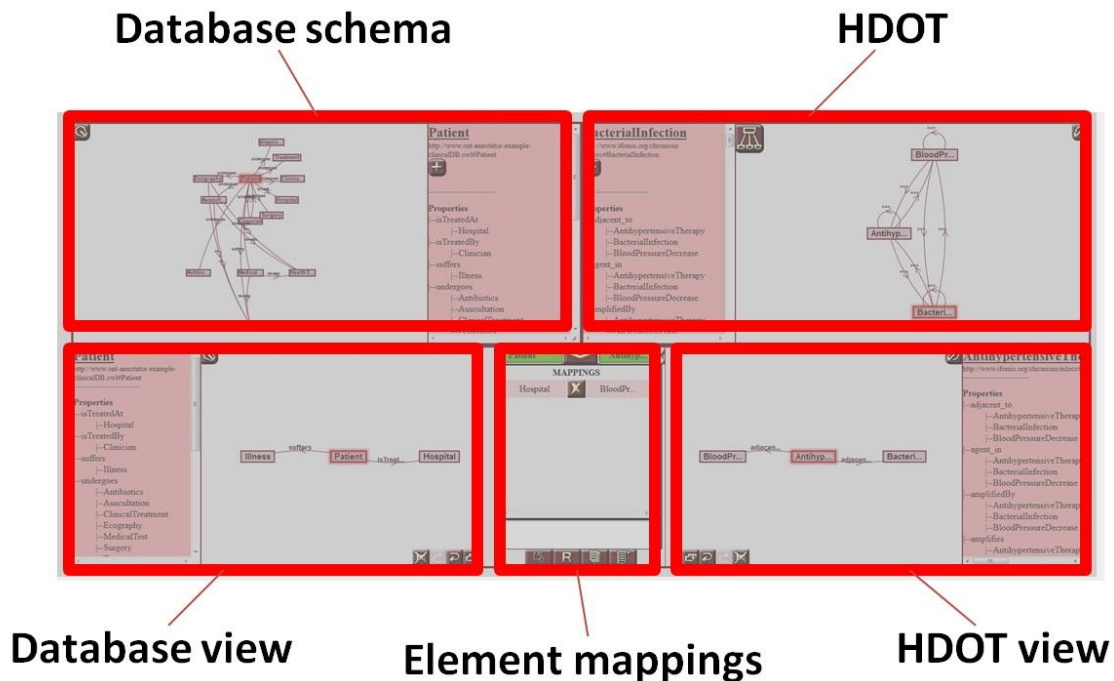

As it can be seen, the OA is divided in five different canvases, as described in the following subsections. Each canvas includes several buttons for controlling the visualization of the information. Users can zoom in and out, and move the panels in every direction.

Database schema canvas

This canvas represents the user database schema as a graph of nodes. In the given graph, **nodes (rectangles) represent classes of the schema**, and **arcs represent relations** between classes of the schema. The graph representation is the most suitable for RDF schemas, given the structure of this type of schemas (classes and unidirectional relations linking those classes). However, when there is more than one relation for the same arc (i.e. more than one relation share the same classes as domain and range), a *** is displayed. The next screenshots show detail of these situations.
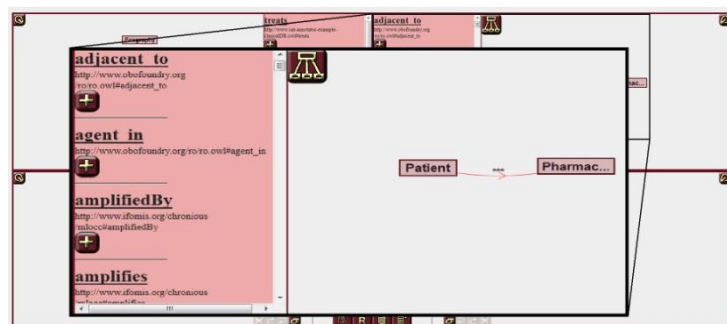
One relation that links the class Patient with the class Clinicial.
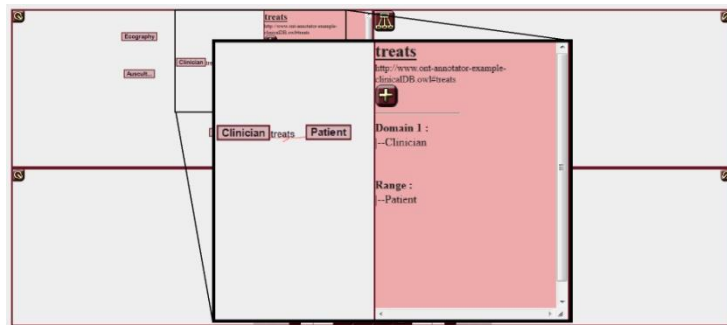
When this relation is selected, information about it is displayed. We can see its domain and range.
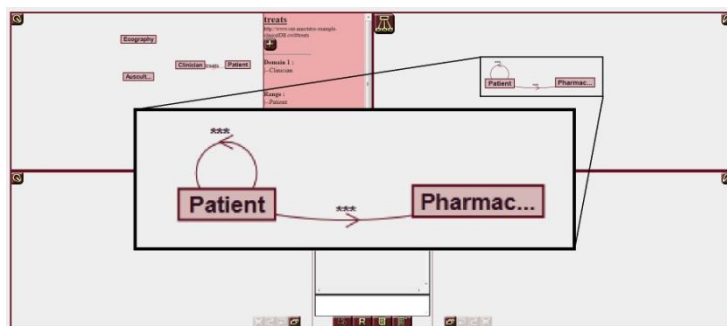
The image shows the case when there is more than one relation linking two classes. A *** symbol is displayed.

When a multiple relation is selected, information about all the actual relations is displayed.
HDOT canvas

This canvas displays **HDOT**, the ontology used as global schema in the p-medicine infrastructure. This canvas also employs a graph for representation purposes. However, there is one important difference between this and the previous case. In most cases, the user database schema will nicely "fit" in its canvas. With HDOT, on the other side, we are facing a structure containing hundreds of classes and relations. Displaying all this content on the screen would simply overwhelm the user, making the search of elements in HDOT incredibly difficult. For this reason, the OA adopts a specific approach for representing HDOT. In the beginning, the canvas shows no classes at all. We can, by clicking on the button on the top-left corner of this canvas, search for classes of HDOT by their name. We can also navigate through the HDOT structure by selecting the parent or children classes of a specific class. When one of these classes is clicked, it is added to the HDOT canvas. This way, **the user is able to build a subset of HDOT** which serves better for his purposes. For example, is the user's database contains clinical information, with patients' information and their treatment details, he would select the corresponding classes of HDOT and build a subset of HDOT with only those concepts. This subset **can be grown at any time**, if the user finds that he needs more classes to complete the annotations.

Even the available classes in HDOT can be customized. HDOT is organized in different **modules**, each dedicated to a specific medical area. Users can click on the 'MODULES' button to select/deselect which of the HDOT modules they want to work with (many modules can be active at any time).

After the user is satisfied with the built subset of HDOT, all that remains is a graph of classes and relations, similar to the graph in the database schema canvas. The user can handle it the same way as the database schema, zooming, moving elements and displaying detailed information of the available elements.

Views canvases

The bottom left and bottom right canvas allow the creation of views from the database and HDOT respectively. By moving elements from the database canvas to the database view canvas, and from the HDOT canvas to the HDOT view canvas, we define the annotation of one single view (which we call "entry"). A project maintains several entries, one of which is the "active" or "working" entry (the one currently being displayed). The controls related to the project entries (save, save as, list, new) are located in the bottom middle of the OA.

To move an element to one of the view canvases, simply select the element in its original canvas and click on the '+' button. To move more elements to the view window, repeat this operation, or simply select the recently added element and click on one of the '+' buttons that it will show (thus adding a relation to another element). The sequence of screenshots below show how to create a view of the database with a few clicks:

Element mappings canvas

Once the two equivalent views are built, individual links (called mappings) between the single elements of the two views must be established. To do this, select one element of each view and add them to the element mapping canvas, as seen on the screenshot below
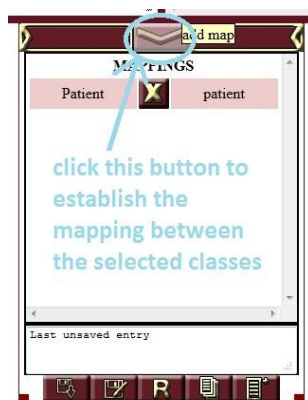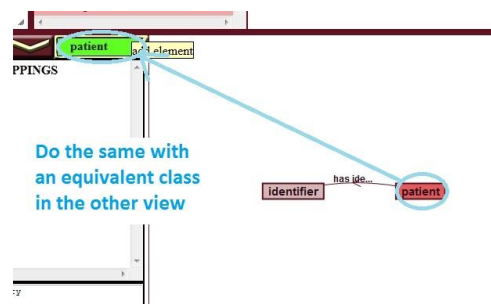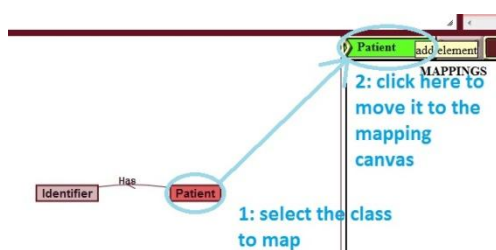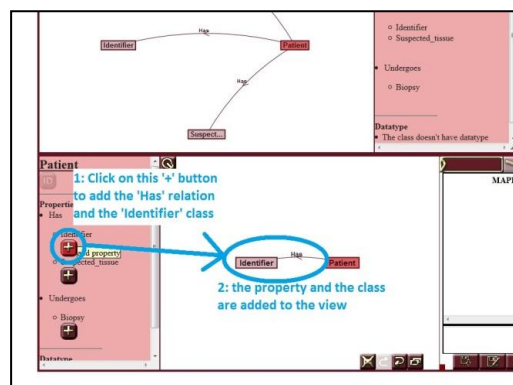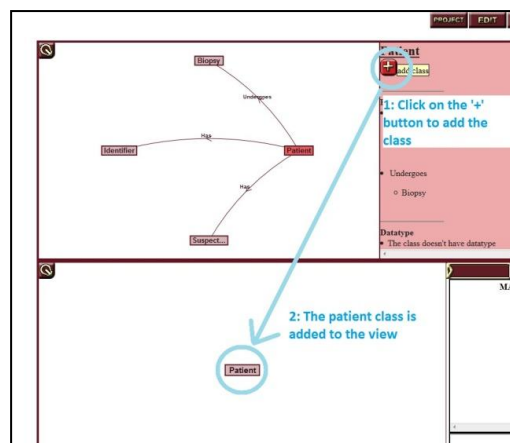
All controls in the OA include tooltips to facilitate its use and comprehension. For more details on the OA and the annotation process, please refer to http://atlas.ics.forth.gr/pMedicine/wiki/index.php/Ontology_Annotator.

12 Appendix E

Task of the external user (user experience designer) for testing the Ontology Annotator

The usability task involves annotating one example database. This database is quite simple, as it provides just a few attributes about patients of a clinical study. The task requires the alignment of these attributes with equivalent attributes in the ontology libraries. The steps of the task are outlined below.

1. Create a new project in the OA

Access the OA tool by login into the p-medicine portal, and create a new project. For the name of the project, set "Example exercise annotation", for the description, set "An annotation example for the usability tests". Select the "example exercise" as database for this project, and select RDF/OWL as the type of the database.

2. Select the appropriate ontology library modules

Elements in the p-medicine ontology library are organized along "modules". For this exercise, the modules which we are interested in are "HDOT core" and "HDOT KLt". Select both of them and deselect all other modules.

3. Add the classes from HDOT

Add the classes from HDOT that you consider you will need during the exercise. These should be equivalent classes to the attribute of the database we are annotating (therefore, you should search for equivalent classes for "Patient", "Identifier", "Biopsy", and "Suspect_tissue"). **TIP**: the most appropriate classes in the ontology libraries are "patient", "neoplasm", "identifier" and "surgical procedure".

4. Create a view of the database which represents the concept "Patient identifiers".

Use the 'Has' property for this purpose. Therefore, the resulting view should be "Patient Has Identifier".

5. Create an equivalent view from the ontology libraries.

For this, you will have to select appropriate classes from the ones selected in step 3 and create a view with them which represents the idea of "patient identifiers". There are many possibilities for creating a view with two classes, since there are many relations between each pair of concepts. Select the relation that you feel is more convenient in this case. **TIP**: there is a "has identifier" relation available for this case.

6. Establish individual mappings between the elements of the created views.

Each class or attribute in the physical view should be linked to one class or attribute in the ontology view. Find these relations and set them. **TIP**: in this case, the links should be "Patient" with "patient", and "Identifier" with "identifier".

7. Give a description text for the generated entry.

Use the "Patient's identifier" text to describe this entry.

8. Save the entry and create a new one for the patients' biopsies and another one for the patients' suspect tissues.

TIP: in the first case, the database view should be "Patient Undergoes Biopsy" and the HDOT view should be "patient participates in at some time surgical procedure". In the second case, the database view should be "Patient Has Suspect_tissue", and the HDOT view should be "patient has part neoplasm".

9. Close the project to finalize the task.

13 Abbreviations and acronyms

ObTiMA	Ontology based clinical Trial Management Application
OA	Ontology Annotator
ISO 9241	Ergonomics of human-system interaction
ISO 9241 – Part 11	Guidance on usability
ISO 9241 - Part 110	Dialogue principles
ISO 9241 - Part 12	Presentation of information
HDOT	Health Data Ontology Trunk
RDF	Resource Description Framework
W3C	World Wide Web Consortium

14 References

ISO 9241 Ergonomics of human-system interaction – Part 11: Guidance of Usability

ISO 9241 Ergonomics of human-system interaction – Part 12: Presentation of information

ISO 9241 Ergonomics of human-system interaction – Part 110: Dialogue principles

Leitfaden Usability; available on DAkkS website (German's National Accreditation Body (former DATech) only in German language) http://www.dakks.de/sites/default/files/71-SD-2-007_Leitfaden%20Usability%201.3.pdf