Privacy and Identity Management in Europe for Life

D4.1.4 High-level Prototypes

Editors:	Cornelia Graf, (CURE)
	Peter Wolkerstorfer, (CURE)
	Erik Wästlund, (KAU)
	Simone Fischer Hübner, (KAU)
	Benjamin Kellermann, (TUD)
Reviewers:	Harald Zwingelberg, (ULD)
	Katrin Borcea-Pfitzmann, (TUD)

Identifier:	D4.1.4
Type:	Deliverable
Class:	Public
Date:	August 27, 2010

Abstract

This document provides an overview of various high-level prototypes developed within the first 29 month of PrimeLife work package 4.1. In general, we will only present prototypes, which were not already presented in other deliverables or heartbeats before. Therefore, this deliverable holds the privacy-enhanced backup prototype, privacy-enhanced event scheduling, and credential selection. The focus concentrates mainly on the different design processes we used for the development of the prototypes.

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 216483 for the project PrimeLife.

Members of the PrimeLife Consortium

1.	IBM Research GmbH	IBM	Switzerland
2.	Unabhängiges Landeszentrum für Datenschutz	ULD	Germany
3.	Technische Universität Dresden	TUD	Germany
4.	Karlstads Universitet	KAU	Sweden
5.	Università degli Studi di Milano	UNIMI	Italy
6.	Johann Wolfgang Goethe – Universität Frankfurt am Main	GUF	Germany
7.	Stichting Katholieke Universiteit Brabant	TILT	Netherlands
8.	GEIE ERCIM	W3C	France
9.	Katholieke Universiteit Leuven	K.U.Leuven	Belgium
10.	Università degli Studi di Bergamo	UNIBG	Italy
11.	Giesecke & Devrient GmbH	GD	Germany
12.	Center for Usability Research & Engineering	CURE	Austria
13.	Europäisches Microsoft Innovations Center GmbH	EMIC	Germany
14.	SAP AG	SAP	Germany
15.	Brown University	UBR	USA

List of Contributors

This deliverable has been jointly authored by multiple PrimeLife partner organisations. The following list presents the contributors for the individual parts of this deliverable.

Chapter	Author(s)
Executive Summary	Cornelia Graf (CURE) Peter Wolkerstorfer (CURE)
Introduction	Cornelia Graf (CURE) Peter Wolkerstorfer (CURE)
Privacy-Enhanced Backup Prototype	Cornelia Graf (CURE) Peter Wolkerstorfer (CURE)
Privacy-Enhanced Event-scheduling (Dudle)	Benjamin Kellermann (TUD) Cornelia Graf (CURE) Peter Wolkerstorfer (CURE)
Credential Selection	Erik Wästlund (KAU) Simone Fischer Hübner (KAU)
Conclusion	Cornelia Graf (CURE) Peter Wolkerstorfer (CURE)

Executive Summary

This document provides a description of high-level prototypes, which were developed in PrimeLife. The development of privacy-enhancing technologies for protecting users' privacy over her lifetime belongs to the main activities in PrimeLife.
The objective of PrimeLife is bringing sustainable and user-controlled privacy and identity management to future networks. For supporting users making informed decisions about the release of their personal data, it is necessary to develop informative, intuitive, legally compliant and understandable User Interfaces (UI) for PETs. Furthermore, users should be able to trust the PETs, from technical point of view, as well as psychological perspective.
The main goal in Activity 4 is providing HCI (Human Computer Interaction) knowledge to other PrimeLife activities to increase usability and user experience when working with the developed technologies and interfaces.
The presented high-level prototypes provide on the one hand insight into currently existing prototypes where work package 4.1 was involved during development; on the other hand, we describe the origin process of the user interfaces. Furthermore, we decided to describe only prototypes, which were not already presented in other deliverables or heartbeats before.
Due to this last constraint, we will present following three prototypes and their design process:

- Privacy-enhanced backup prototype, which enables the delegation of access right
- Dudle, a privacy-enhanced event-scheduling and
- Credential Selection, which provides a selection mechanism for private data release.

Other high-level prototypes of Activity 4 like data track or policy interfaces were already presented in D4.2.2 and D4.3.2.
The first Chapter of this document briefly introduces the background the background of the user interface design within PrimeLife and the structure of this deliverable.
Chapter 2 describes the UI-development of the privacy-enhanced backup prototype. This prototype development takes place in two main parts, first the design of mock-ups, which were animated with flash. Second was the design of an HTML-frontend for the prototype.
In Chapter 3 we present a privacy-enhanced event-scheduling tool called Dudle. This tool allows to schedule events or create pools. Furthermore it provides the possibility to vote anonymously.
The last prototype, credential selection, is presented in Chapter 4. This prototype allows a user e.g. to selectively reveal only a subset of her attributes.
The final chapter provides a comparison of the three different design processes, which were used for the development of the UIs.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The usage of privacy-enhancing technologies (PETs) is an important factor for lifelong privacy protection. PETs, which are currently developed in PrimeLife, can be very powerful tools for users to control and track the release of their private data in web. Anyhow, one main part when designing PETs is to present the complex techniques of PETs in an understandable way to end-users. Only if the functionality of PETs is understandable for end-users they will be able to work with them and protect therefore their privacy in an active way.

Therefore, one main task when working on PETs is, to create the user interfaces (UI) in a way, which prepare the complex technical concepts of PETs for end-users. In addition, end-users need to be able to use UI elements and workflows not only in an understandable, but, furthermore, in a familiar way. This is utterly important because end-users will only accept usable applications and therefore utilize them in their daily live. Therefore, one main tasks of PrimeLife Activity 4 is to provide HCI (human computer interaction) knowledge to other activities within PrimeLife with the task of creating useable PETs.

In this deliverable, we will provide an overview of the UI of three high-level prototypes, which were developed in work package 1 of Activity 4.

- The UI of the first presented prototype, the *privacy-enhanced backup,* was realised in cooperation between Work package 4.1 (Front-end) and Activity 1 (Middleware and Back-end). This prototype is described in detail as at the beginning of the design phase, no technical requirements had been defined and therefore the HCI-experts had to partially define the requirements by designing the prototype. Prototyping in this case is seen as RE (requirements engineering) technique.
- The deliverable also describes the UI for the *privacy-enhanced event-scheduling* tool – called Dudle. During the design of this prototype, the HCI-expert and the tool developer worked in the same room and had had continuous exchange of ideas during the creation of the application's UI. Furthermore, the HCI-expert gave daily feedback to the developer.
- We also present the prototype on *credential selection,* which offers the user a selection mechanism by which she can decide which of her private data shall be released. This prototype was designed in an iterative design process with end users.

In the following three chapters, we will introduce the prototypes mentioned above and describe the design processes we used. The last chapter summarizes the design process and provides an overview of benefits and disadvantages of each process.

Part of the work reported in this deliverable is still in progress, so the usability test for the privacy-enhanced backup prototype and the privacy-enhanced event scheduling were not done yet. Results of these tests will be reported in the final HCI research report D 4.1.5, which will be published in the end of PrimeLife project.

Chapter 2

Privacy-Enhanced Backup Prototype

The following chapter describes the development process of the frontend of the privacy-enhanced backup prototype from conceptual design until realization in HTML.
Within the report of the first EC project review, one of the reviewers stated that: *"It is expected that HCI research will drive the implementation of relevant functionalities in the Focal Demonstrators."*
Accordingly, we opted for a design process that puts the development of the UIs in the first place before the prototype backend is implemented. This design method intends to make the output of the HCI-related project-parts of PrimeLife more visible.
When we started to work on the user interface design of the privacy-enhanced backup prototype, technical requirements had not yet been specified. Because of this, the designers had plenty of challenges to tackle when creating the UI. But, that way, it was possible for us to focus entirely on the actual user needs and user requirements by ignoring any technical aspects of the prototype. Using prototyping as *requirements engineering* (RE) method enabled us to "drive the implementation of relevant functionalities" of the privacy-enhanced backup prototype like demanded in the technical report.
The mock-ups have been created using the open-source GUI-prototyping tool pencil. We applied the following process: first, we created mock-ups of certain functionality; followed by an interactive prototype to collect feedback and input from the prototypes developers. The advantage of such a proceeding is that fundamental questions arising from UI design could be clarified in a very early stage of development. An example for such a fundamental question is the following: is single user access for an Area of Life[1] (AoL) enough, or should it be possible to give access to a group of users?
This question-answer-communication and the comments given on the created mock-ups resulted in getting them more and more mature over time. Finally, we have been able to release a clickable high-level UI prototype, which simulates functionality in such a way that we can evaluate it with end-users. This UI prototype does not provide any implemented functionality and serves to increase understanding and communication between designers and developers, in the first place.

[1] An AoL in respect to the delegation prototype refers to a set of data chosen by the delegator which refers to a certain aspect of life such as workspace, insurance information and tax data. These AoL may be required by colleagues in case of unexpected absence or by heirs and children in case of death of the delegator. Cf. [5]

This first UI prototype enables us to uncover problems and to work on new ideas to improve the underlying concepts.

2.1 High-level Prototype for Privacy-Enhanced Backup and Synchronisation

This section deals with the high-level UI prototype for privacy-enhanced backup and describes the idea behind it.

The prototype was the first outcome of our iterative design process. It merges the usability knowledge of the HCI experts with the first specified requirements of developers. Please note, that this prototype do not deal with partial identities.

In the beginning of prototyping, we wanted to provide the possibility of interacting with the mock-ups. As a first approach, we animated our mock-ups using Flash. Therefore, we got a prototype, which did not need any source code to program but provided interactivity of some sense. This was done to illustrate the interaction possibilities and to get insights into possible interaction problems.

To provide an easy-to-understand interface we decided to go for a tripartite solution (Figure 1).

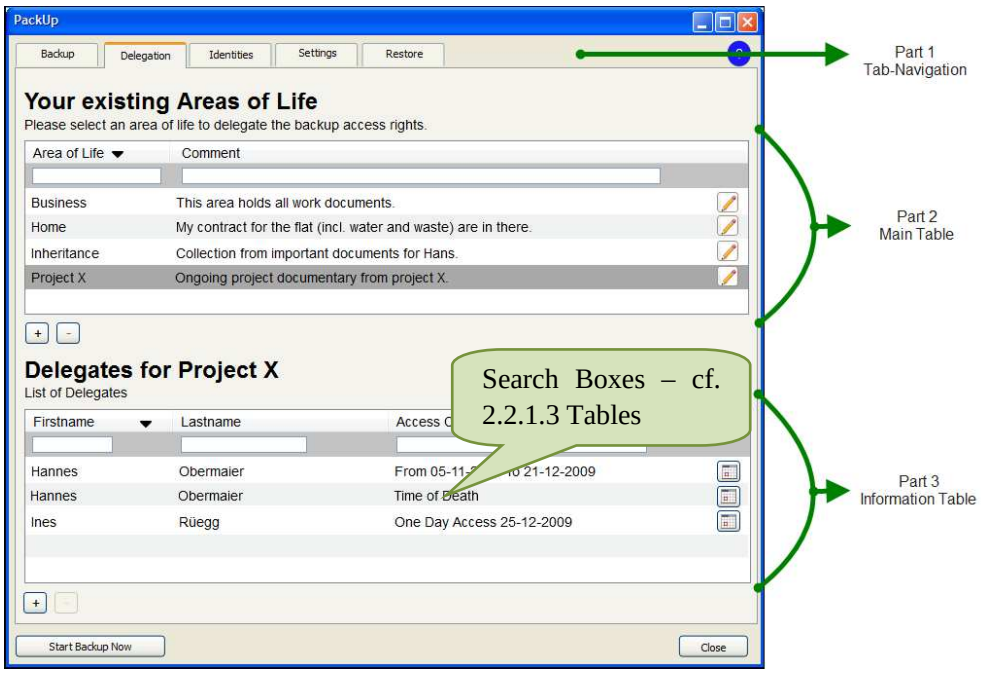

Figure 1: Structure of the MockUp

UI - Part 1: Tab Navigation

The tab navigation consists of the five tabs in the upper part of the window. Using these tabs enables the user to navigate between main options (e.g. delegation, restore).
We decided to apply tab navigation because it is a common means to most users known from web browsers. Furthermore, tabs allow users to recognize on first glance which information will be provided in a section cf. [10].

UI - Part 2 & 3: Tables

Tables are familiar for most users and they know how to read them and how to work with them. Even if a user never has worked before with a table in software, she knows from real life how to read tables, e.g., bus schedules.

Underneath each table, there are a “+”button and a “-“button. These buttons allow users to add new elements to the table or delete existing entries.

UI – Part 2: Main Table

The second part of the window consists of the main table. This table always contains the main information for the tab, e.g. in the Location tab – further information to each location. (cf. Figure 2).

UI - Part 3: Information Table

The information table will only become visible when a user selects an object in the main table. It gives detailed information for the selected object. For the previous example this would be a list of delegates (cf. [5]) for a selected Area of Life (cf. Figure 4).

2.1.1 Backup

The flash-version of the Backup-tab was in a very early iteration draft when we started implementing the HTML-prototype. Therefore, the flash-version of the Backup-tab is unfinished.

In this tab, the locations for the backup storage are managed (Figure 2). The user sees on the first glance in which locations her data referring to a given AoL are stored. Furthermore, the application displays the time of the last and next backup and also the duration of the last backup.

The user may add or delete or edit a storage location, she can change the next point in time for thebackup, and she can adjust an interval for backups.

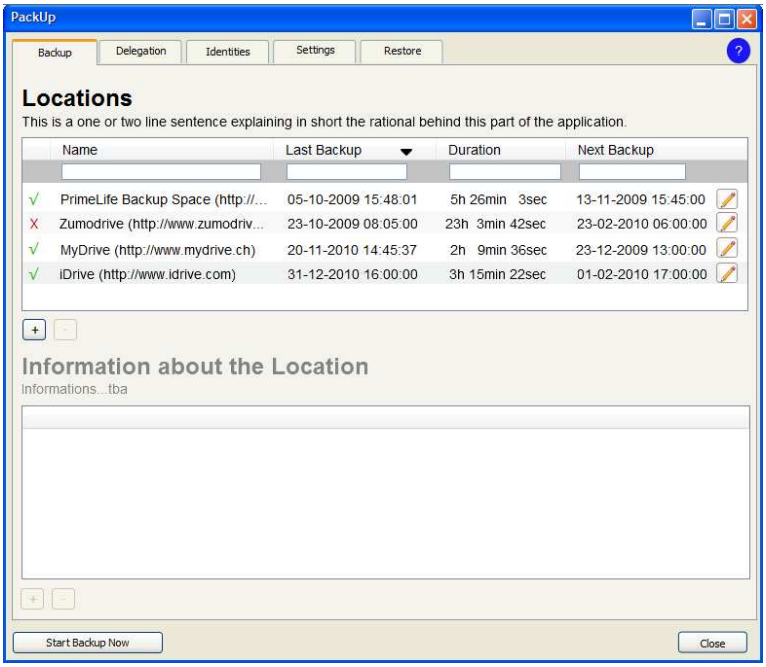

Figure 2: Backup View

2.1.1 Delegation

The Delegation-tab provides all information concerning Areas of Life and the associated delegates. In this section we will present a workflow for this tab.

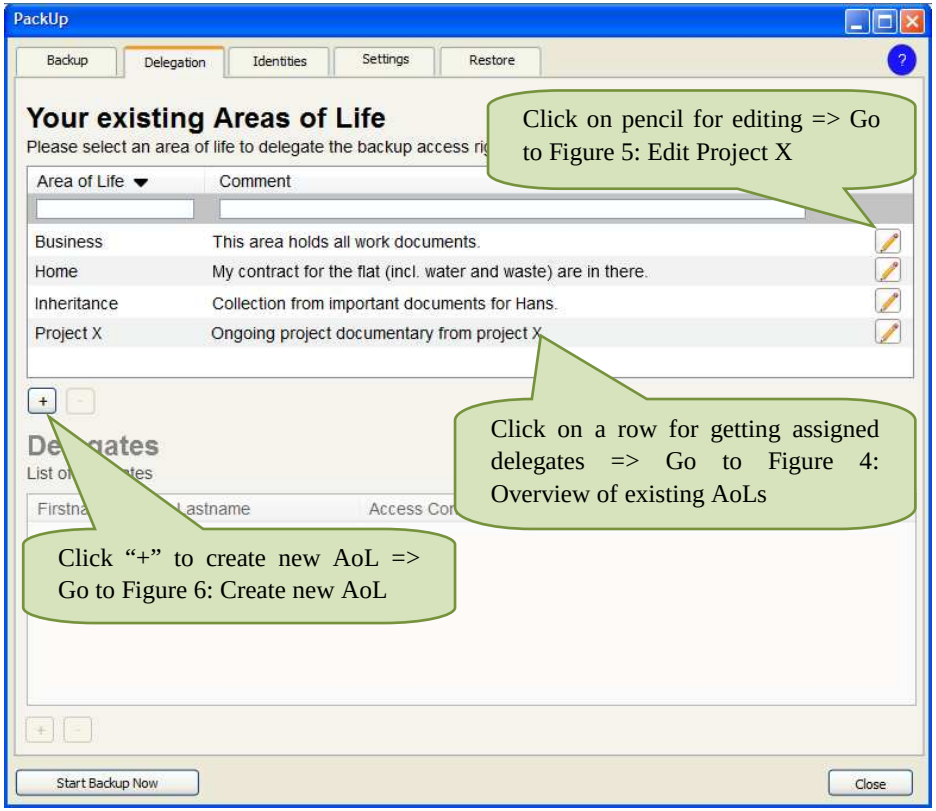

Figure 3: Start screen of Delegation-Tab

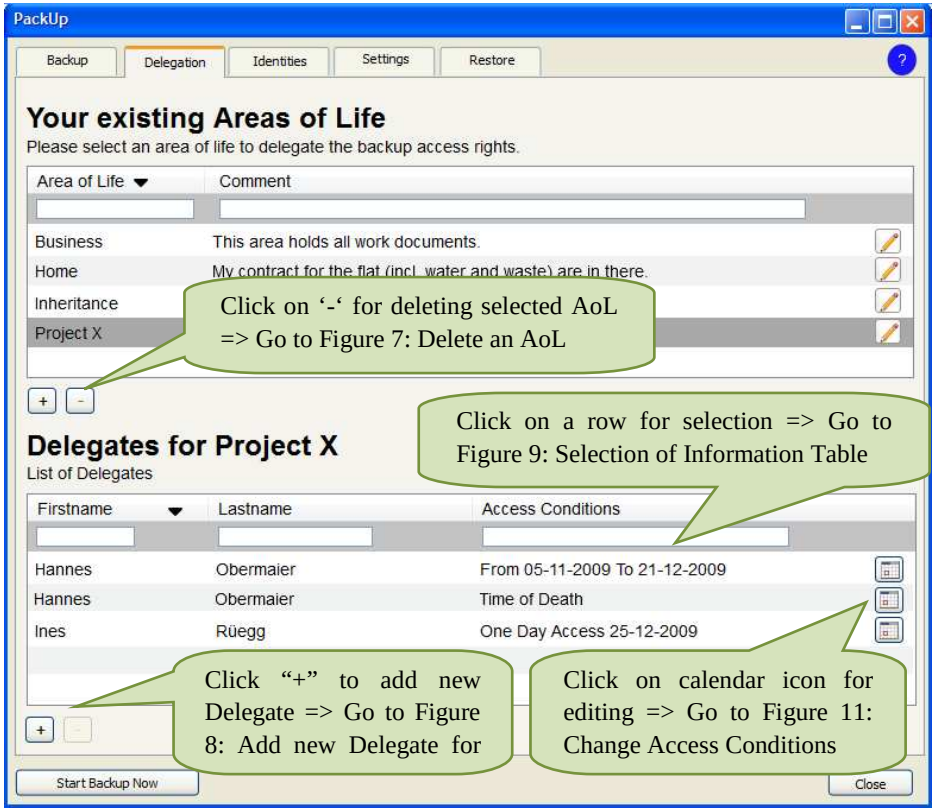

Figure 4: Overview of existing AoLs

This is the start-screen of the delegation tab.

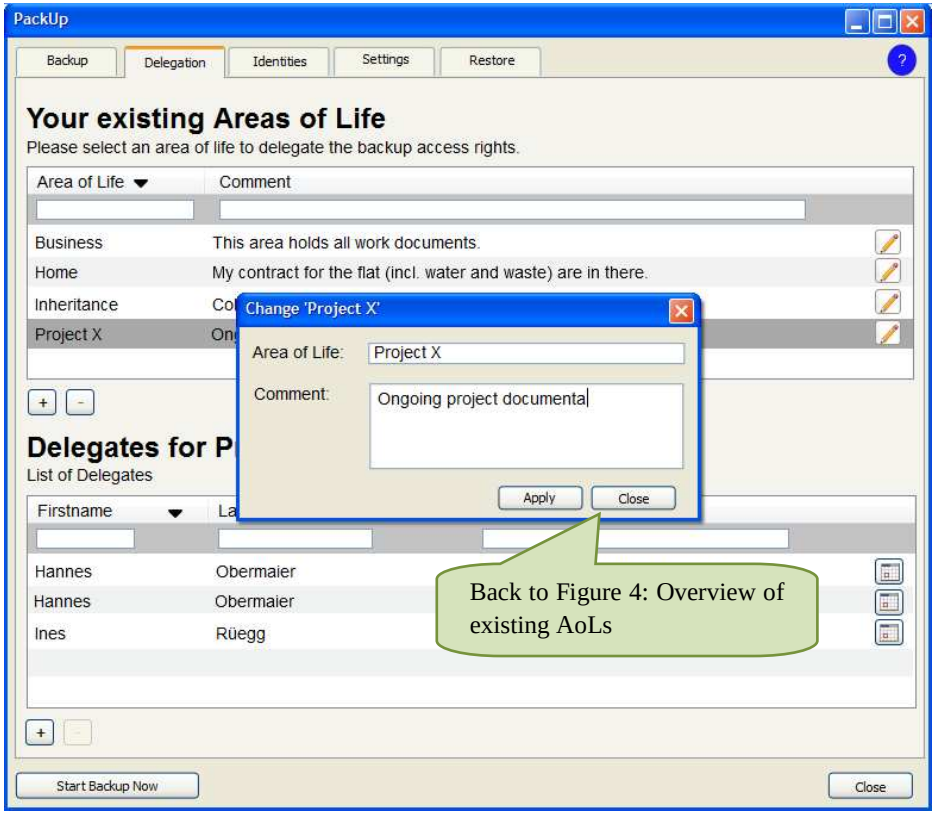

Figure 5: Edit Project X

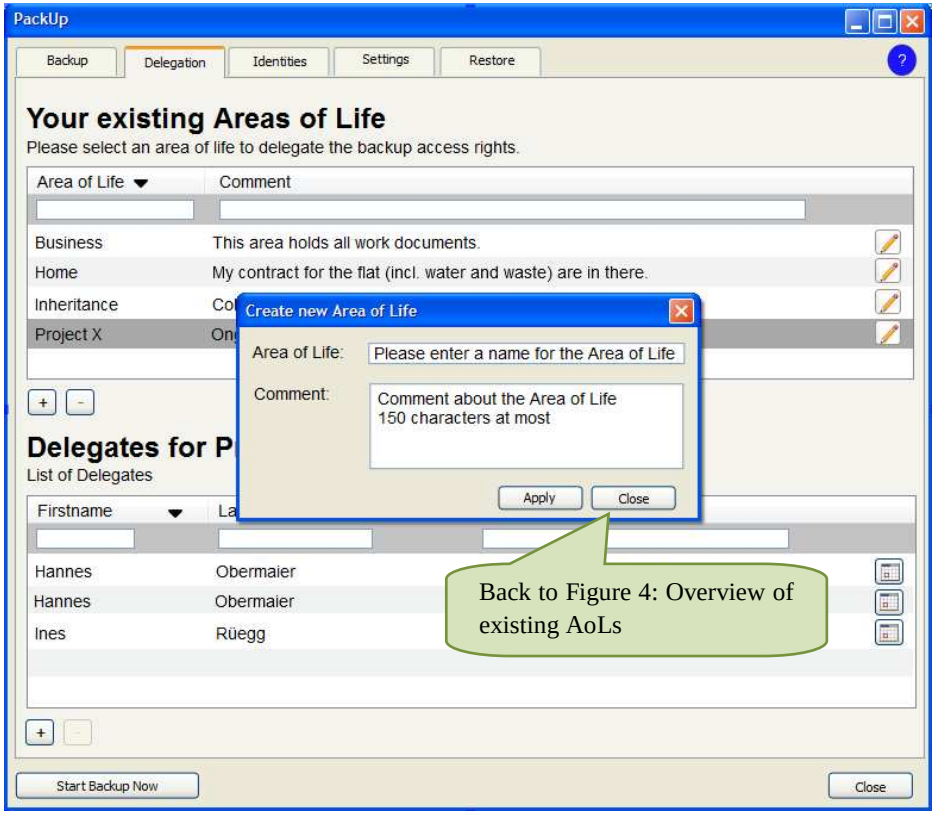

Figure 6: Create new AoL

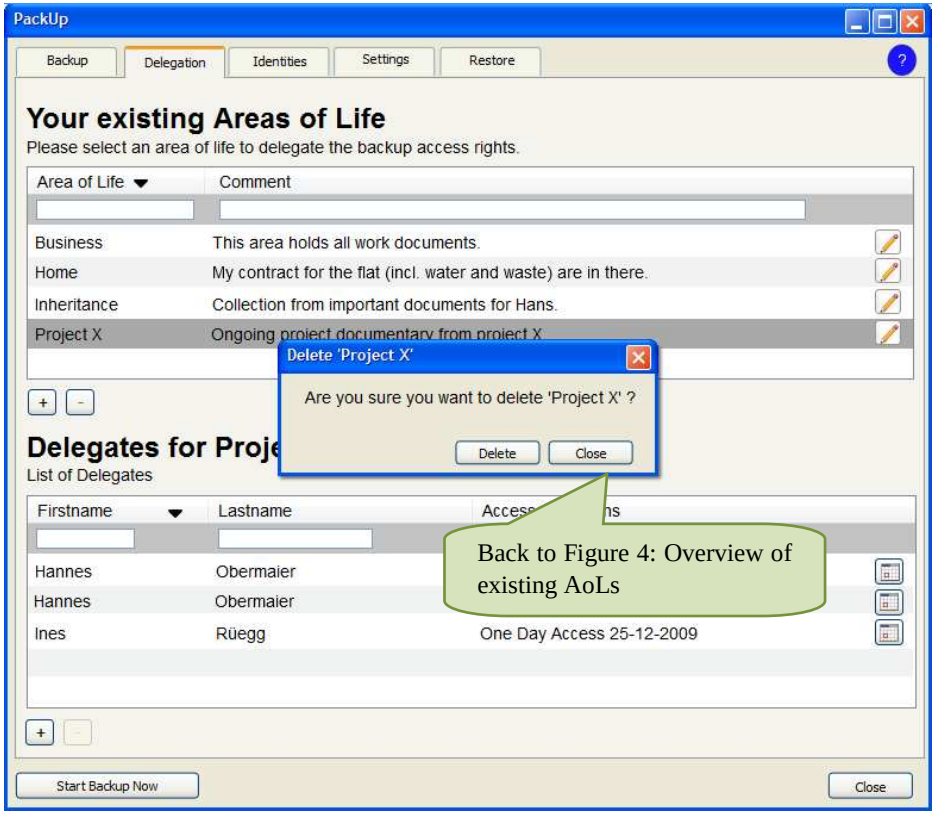

Figure 7: Delete an AoL

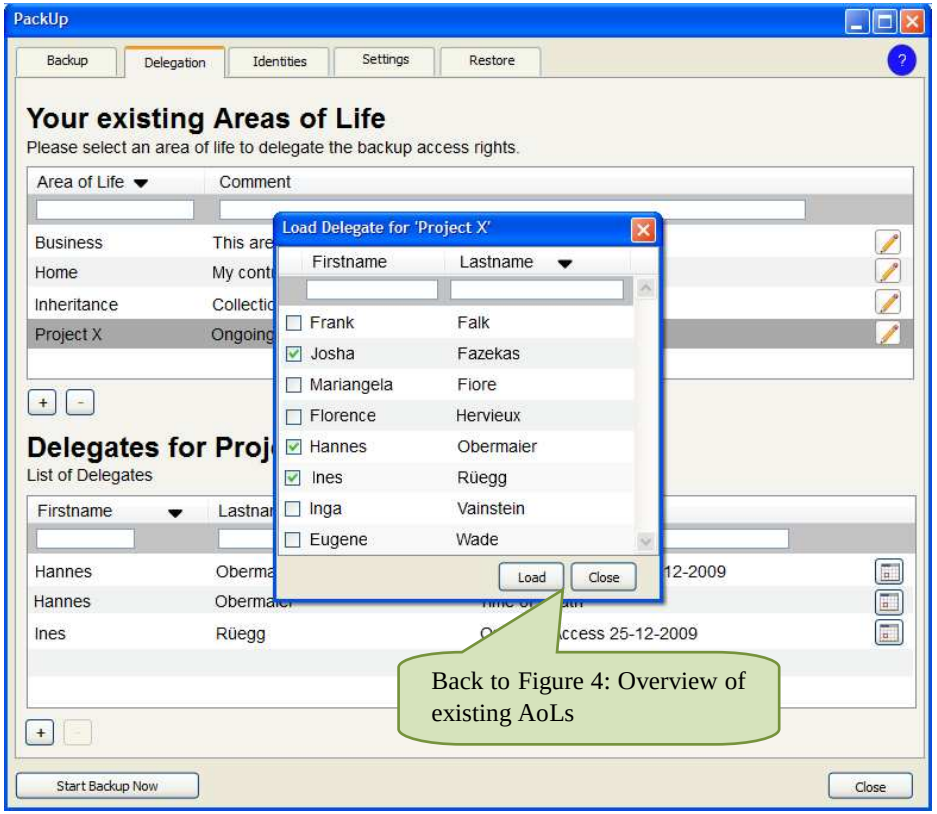

Figure 8: Add new Delegate for Project X

Deleting an Area of Life also deletes the assigned delegates for this AoL.

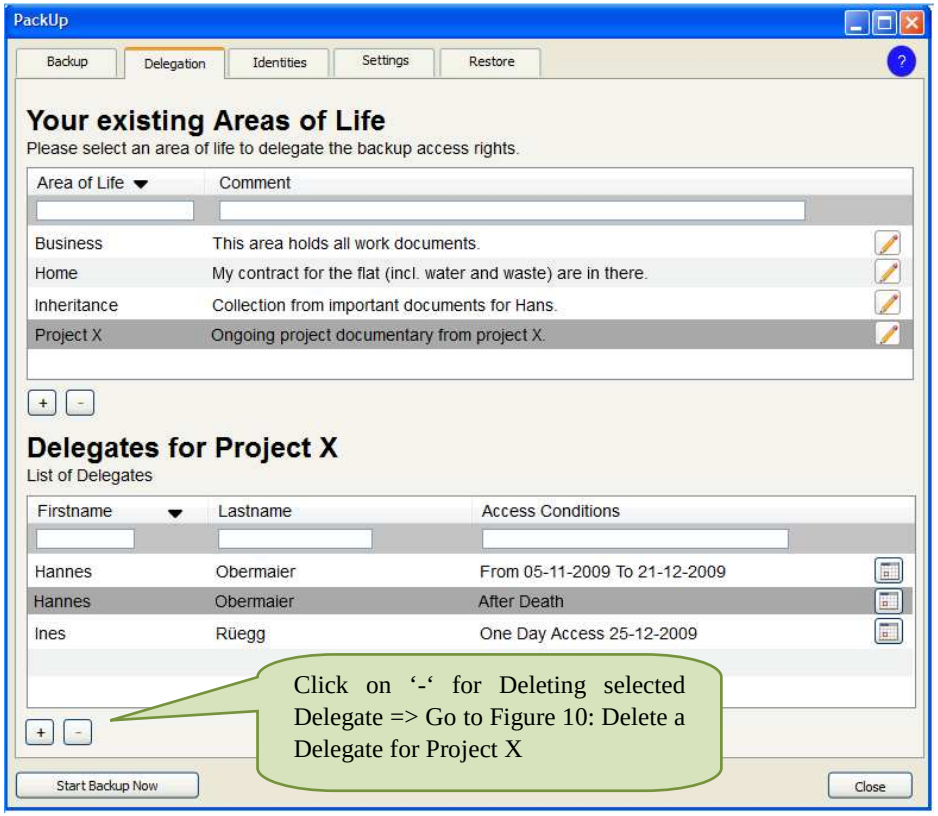

Figure 9: Selection of Information Table

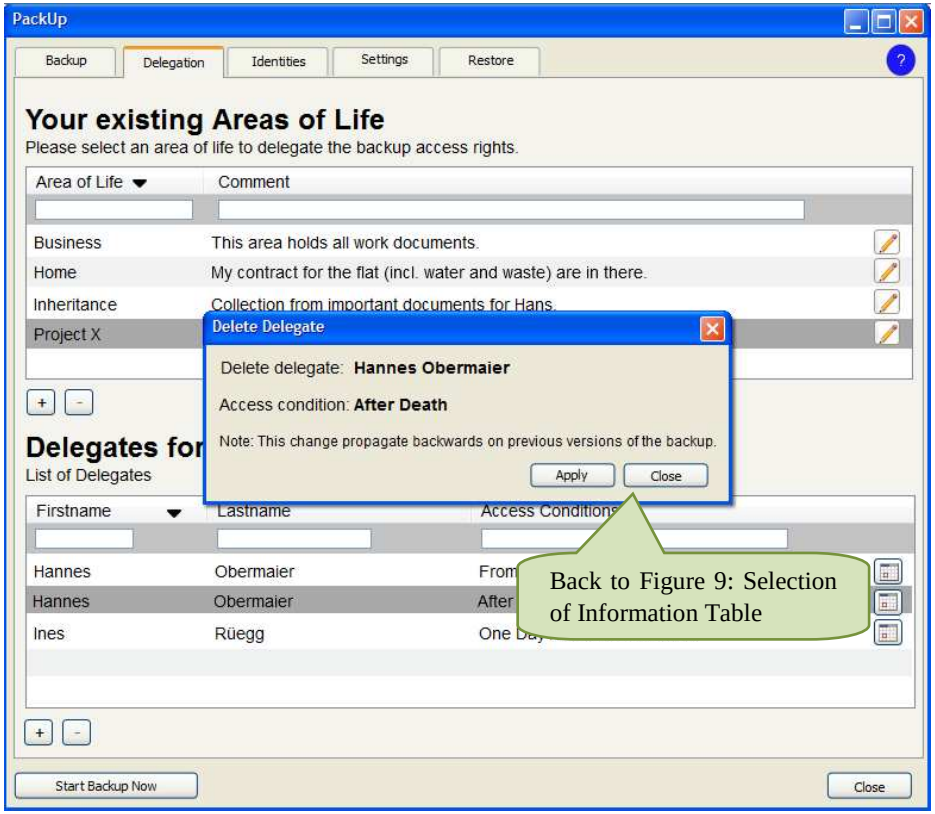

Figure 10: Delete a Delegate for Project X

Deleting a delegation from an AoL has effects on previous versions of this AoL backup.

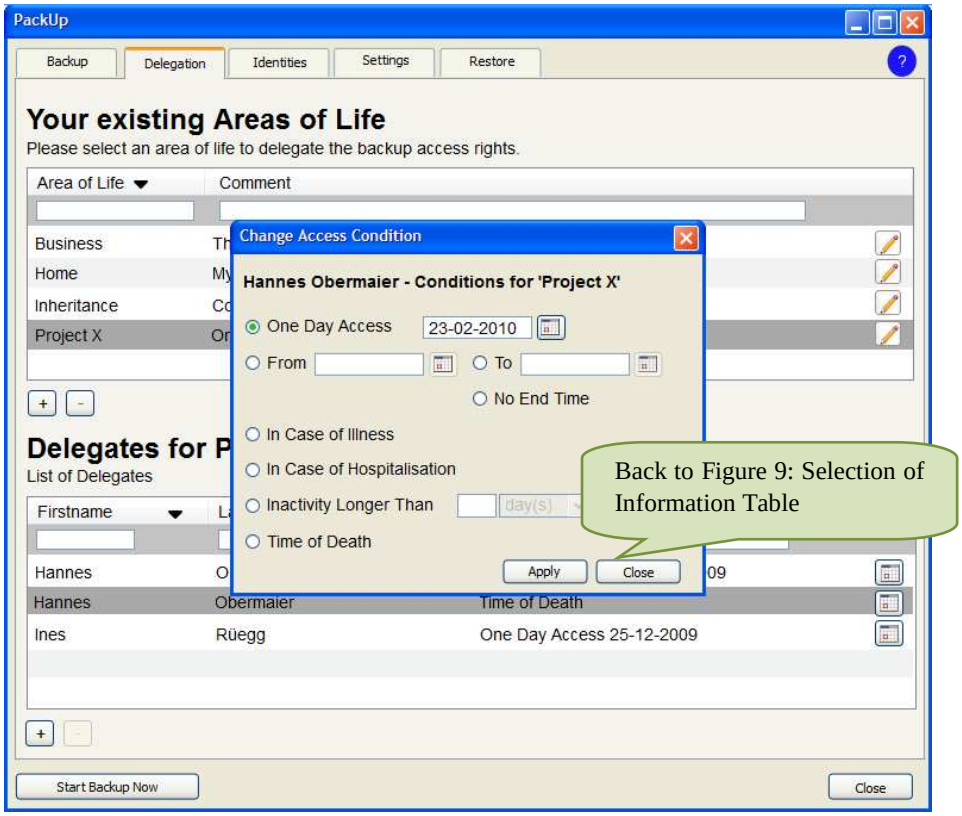

Figure 11: Change Access Conditions

2.1.2 Identities

The Identities-tab provides information about all delegates; furthermore it displays information about assigned Areas of Life (cf. Figure 12).

In general the identities tab and the delegation tab consists of the same content – only the focus is different. The Delegation-tab is driven by the areas of life and the Identities-tab is driven by delegates.

This view shall help users to manage all her delegates, and it allows a quick and easy overview on which AoLs are assigned to a delegate.

If a user clicks on '-' for deleting a delegate, this action has influence on all AoLs the delegate is assigned to.

Removing an assigned AoL from a delegate has also influence on backup-versions of an AoL. This means access to old backups will be denied as well. Keys assigned to the former delegate will be revoked etc.

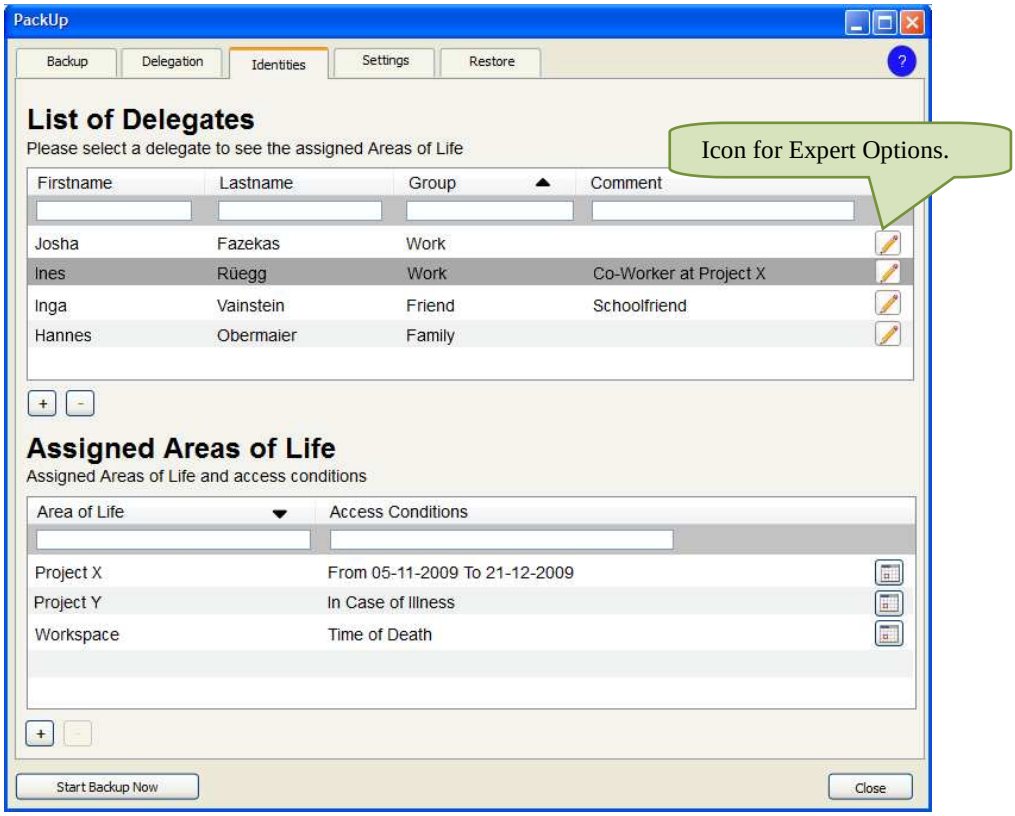

Figure 12: List of Delegates

2.1.3 Settings

In the beginning of design-process, we had the idea that the settings tab should, e.g., allow switching between standard view and expert view. During the design-stage, we decided to put the expert options next to each table-row (cf. Figure 12). That way, the user can decide for each element whether she wants to use expert options or not.

So the tab in the interface is provides no functionality, anyhow we decided to keep it in the mockup design for documentary reasons.

2.1.4 Restore

The Restore-tab provides an overview of all backup versions of an AoLs that are possible to restore (cf. Figure 13). Furthermore, it shows the backup version history for an AoL and supports the recovery of each backup version in the list.

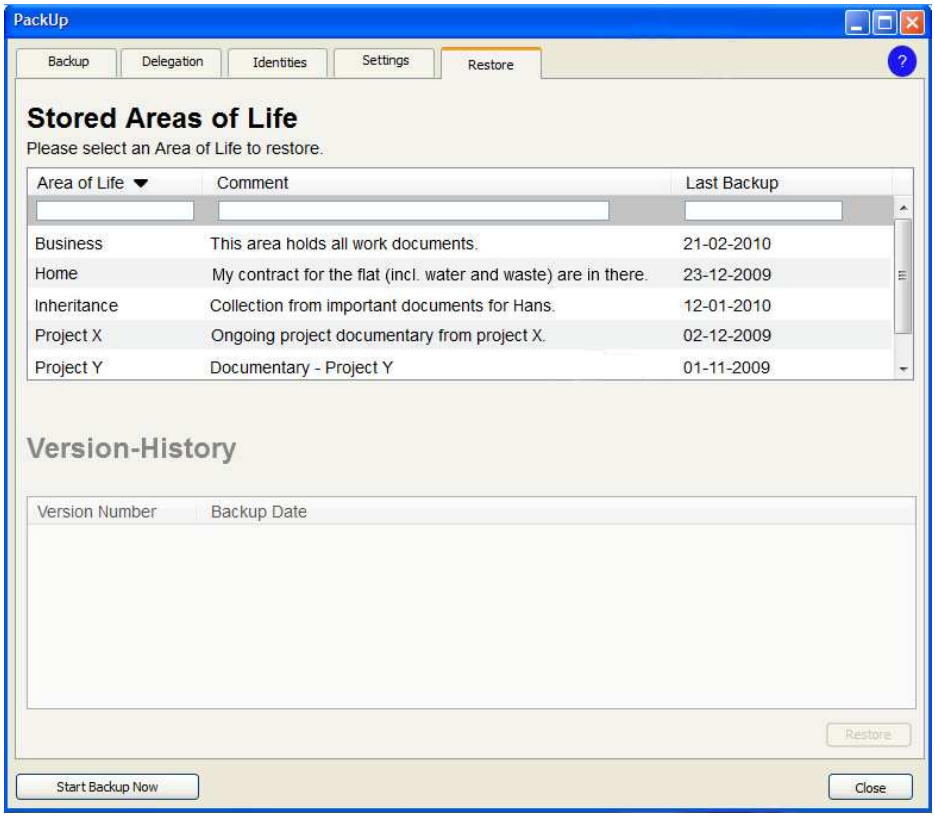

Figure 13: Restore

2.2 HTML based Prototype

During the concept stage of development of the privacy-enhanced backup demonstrator, it was decided to realize that demonstrator as a web frontend and not as a desktop-solution.

Therefore, we started with the front-end development of an HTML-based prototype, which bases on findings of the Flash-prototype. Heartbeat 1.3.6 was released before we started with the development of our HTML-prototype. Therefore, we also include certain requirements specified in Heartbeat 1.3.6 in the HTML-version of the prototype.

Anyhow, the HTML-prototype only implements a standard view. Nevertheless, in Heartbeat 1.3.6 specification for an expert actions are defined and so we create placeholders for an expert view. Therefore, we put an “options” button everywhere where complex interaction could be possible.

The current version of the prototype provides UIs of all basic functionality, which is necessary for using the privacy-enhanced backup tool. In further development, it will be extended by an expert view that allows, e.g., adding and deleting of single files to and from an AoL.

Since the structuring (main navigation, main table, information table) in the first prototype prove its worth we decided to maintain it also in the HTML prototype. The advantage of the structuring is that all necessary information for the different specifications of the prototype can be displayed due it.

Since we made good experiences with the, we decided to maintain the structure, which is illustrated in Figure 14.

The left sidebar is responsible for navigation in the tool. The user can navigate with the drop-down menu between her partial identities (pID) and received delegations. Partial identities are identities of the owner of the privacy-enhanced backup tool, e.g., for work, for private stuff and so

on. Received delegations are stored areas of life from other users, e.g., a co-worker delegates me access to her data in case of illness.

In the centre of the window are two tables. The main table provides the main information for the currently selected navigation entry; the information table gives detailed information for selected objects in the main table. Furthermore, the user can perform different actions in each screen. The executable action depends on the selected navigation entry, e.g. in Delegation the user can manage her areas of life and delegation for each area.

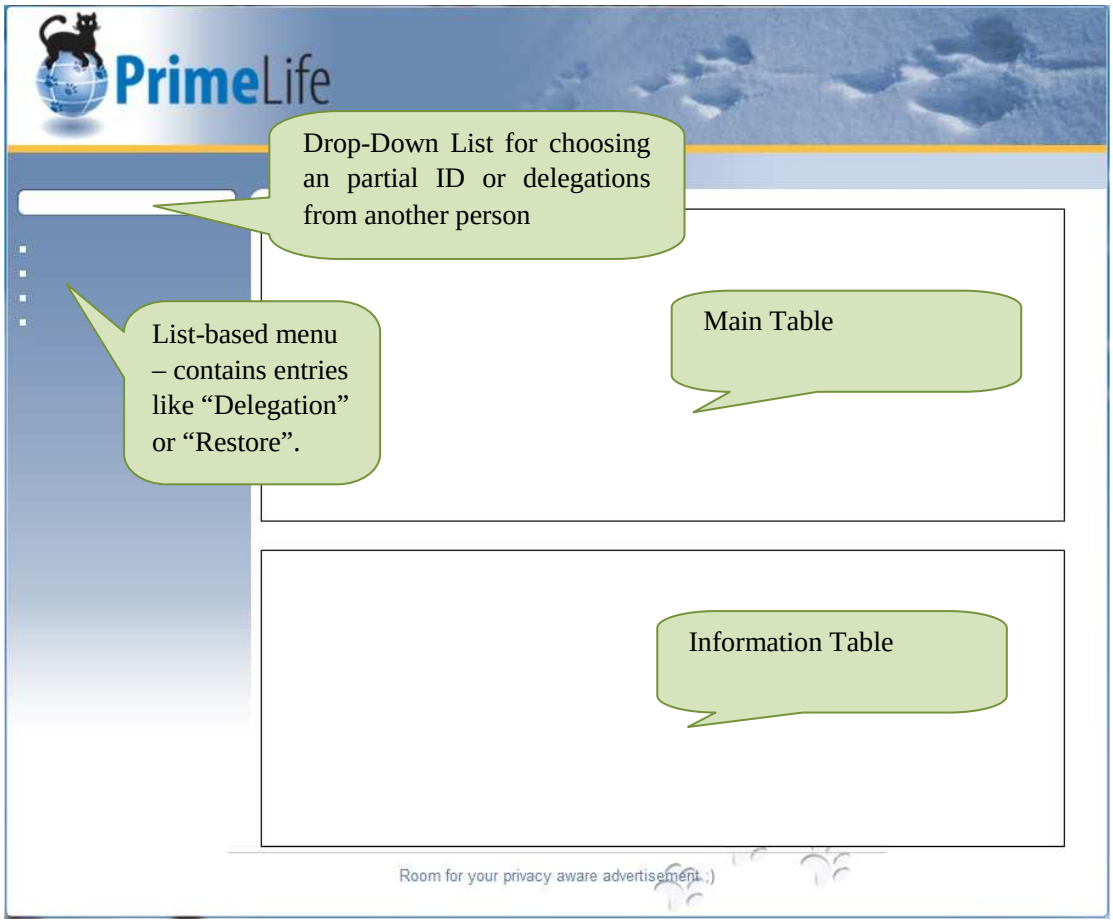

Figure 14: Structure of HTML prototype

2.2.1 UI-Elements and Concepts

To keep the application simple and the learning effect for users low we used only proven and widely used UI elements and concepts.

2.2.1.1 Drop-Down List

The drop-down list in the privacy-enhanced backup prototype enables the user to switch between different pID’s and delegations she got from other persons.

We decided to use this element, because drop-down lists may contain many elements and are still useable. We based our UI on the assumption, that a user may have many pIDs and even more delegations from other persons during her lifetime.

2.2.1.2 List-based Menu

The list-based menu on the left side of the page contains the main navigation points for the UI. Since we differentiate between the delegation view and the pID view, the main navigation points vary in both views. The partialID view contains elements for administration of a pID (Backup, Delegation and so on). The “received delegation view” contains a list of delegation the user got from another person and the possibility for restoring.

Lists are used in many websites for navigation. So, the concept is familiar to most users. We decided to use this list-based menu for the main navigation since it is easily extendable for developers and the cognitive load for users is very low.

2.2.1.3 Tables

Like previously mentioned, tables are familiar for most people – if not from a computer application than from real life, e.g. math-tables in schools.

So we decided to use tables for data representation. The elements in tables are variable; if the table contains more elements than displayable on screen, a scrollbar appears at the right side of the table.

Sort

The columns of the table provide a sorting function; this means that users can sort a column ascending or descending order (cf. table sorter [10]).

Search

Another advantage of the used table is that it provides search functionality, which adjusts the table after each entry of a letter. This method is called “Refining Search”- cf. [7]. For example, if the user enters “H” at the column “First name” in Figure 1, the table will be updated and only the two entries starting with an “H” are displayed.

Alternating Row Colors

To provide better readability of the table content, we use alternating row colors. This means one row is white and the next grey –this striped looking table is called a zebra-table, cf. [10]

2.2.1.4 Buttons

The ‘+’ and ‘-’ buttons underneath each table allow users to add a new element or delete a selected element from the table.

The button “Options”, which appears next to a table row, does not provide any functionality in the current prototype – it is a placeholder for future integration of expert options.

2.2.1.1 ToolTips

Tooltips provide information about what will happen when the user clicks on an interaction element of the software. This is a great advantage because tool tips inform users about consequences of an interaction (cf. [5]).

Figure 15 shows a tooltip for a mouse-over on editing-button. The tooltip will be displayed as white text in a dark-gray box. By using a box the readability of the text is always given, independent which color the UI has.

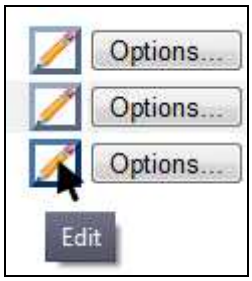

Figure 15: Detail of Editing and Expert Option with Mouse Over

The tooltips will be displayed on mouse over, e.g., when the user moves the mouse over '-'button "Delete selected **name of selected element**" will appear.

2.2.1.2 Pop-Up

We decided to use pop-ups for further interaction possibilities. When we came to this decision, we were aware that pop-ups have a bad reputation because they are often used for unintentional advertisement. Anyhow, pop-ups are a great possibility to provide content just when needed and pop-ups are easy to use and understand even for inexperienced users. One important thing when providing pop-ups in an application is that they need a possibility for easy closing.

Furthermore, we were aware of the possibility that some users block pop-ups in their browser. Anyhow, enabling pop-ups for selected websites is possible.

2.2.2 Start screen

The start screen has only one goal: Users shall select a partial identity or a delegation from another person to get more information about it.

As Figure 16 shows, there are two different possibilities to choose an identity, a drop-down list and radio buttons. In general, it is proscribed to provide two different possibilities for the same interaction, but in this case, it was necessary for consistency reasons.

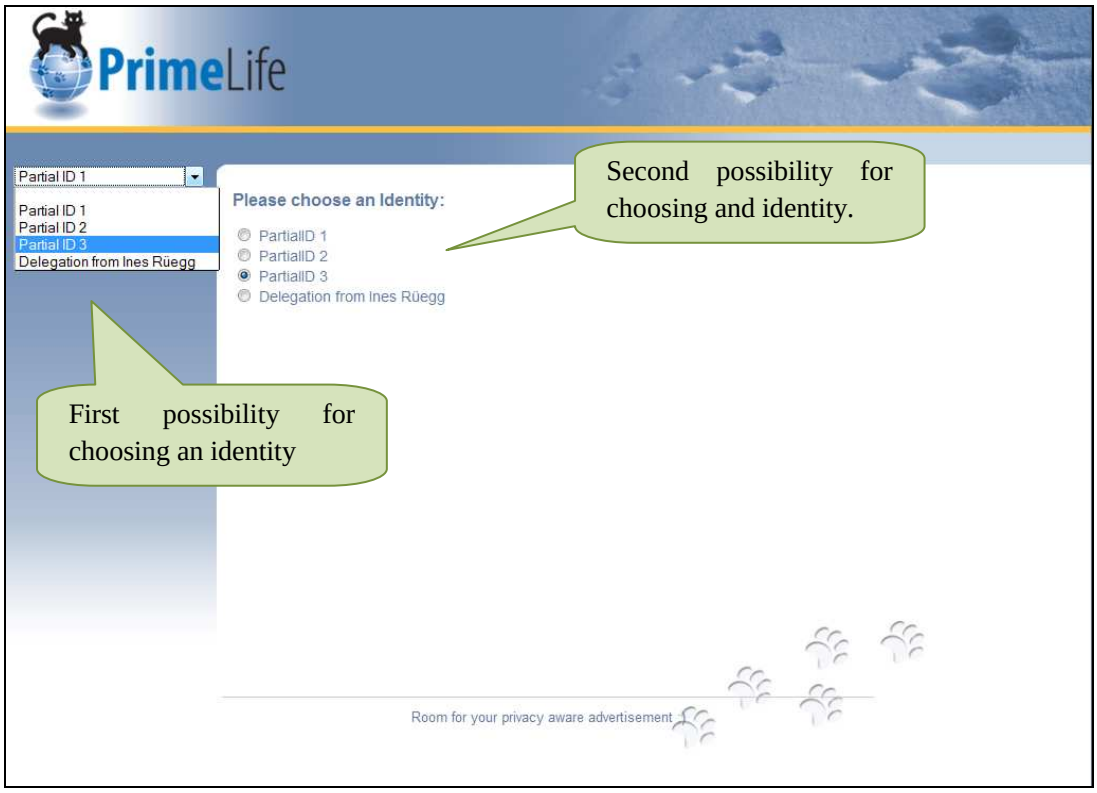

Figure 16: Start screen of the HTML-based Prototype

The first way, the drop-down list, is used throughout the whole application. Therefore, it is necessary to provide it also on the start screen. As second way, we used radio buttons in the main interaction part of the window. This is necessary because the main interaction part always provides the main interaction possibilities for the current view – and on the start screen; this means the choosing of an identity, cf. Figure 16.

We decided to separate the interface into two main parts: first the administration of users partial IDs and second the possibility to manage delegations received from other persons.

This clear splitting of the interface into two main parts reduces complexity for users and allows us a more task-oriented UI design; the UI provides solely necessary information for the current part.

The two parts will be described in the following sections:

- Section 0 describes the possibilities for editing a partial ID.
- Section 2.2.4 describes the interaction possibilities for delegation received from other people the user is currently holding. This is called “delegation view” in the further document.

2.2.3 Part 1: Partial Identity

The pID view provides the administration of a user’s AoLs. It is the control center of a user for editing, delegating, storing and restoring an AoL.

The pID view provides the administration for the following topics concerning the user’s AoLs:

- backup of an AoL,

- overview of the existing AoLs,
- a list of delegations to other people (e.g. family, co-workers...), and
- the recovery of an AoL.

If a user chooses one of her IDs for editing, she will start at the menu point “Delegation” (cf. Figure 18).

2.2.3.1 Backup

Figure 17 shows the backup screen. Here the user gets an overview of all backup storage locations of the chosen pID. For each pID different storage locations may be chosen.

Figure 17: Backup Screen

2.2.3.2 Delegation

Here the functionality for managing existing AoLs or creating new ones is provided. The delegation of an AoL to other people, like co-workers, is also possible.

Figure 18 to Figure 21 present a workflow for the delegation point.

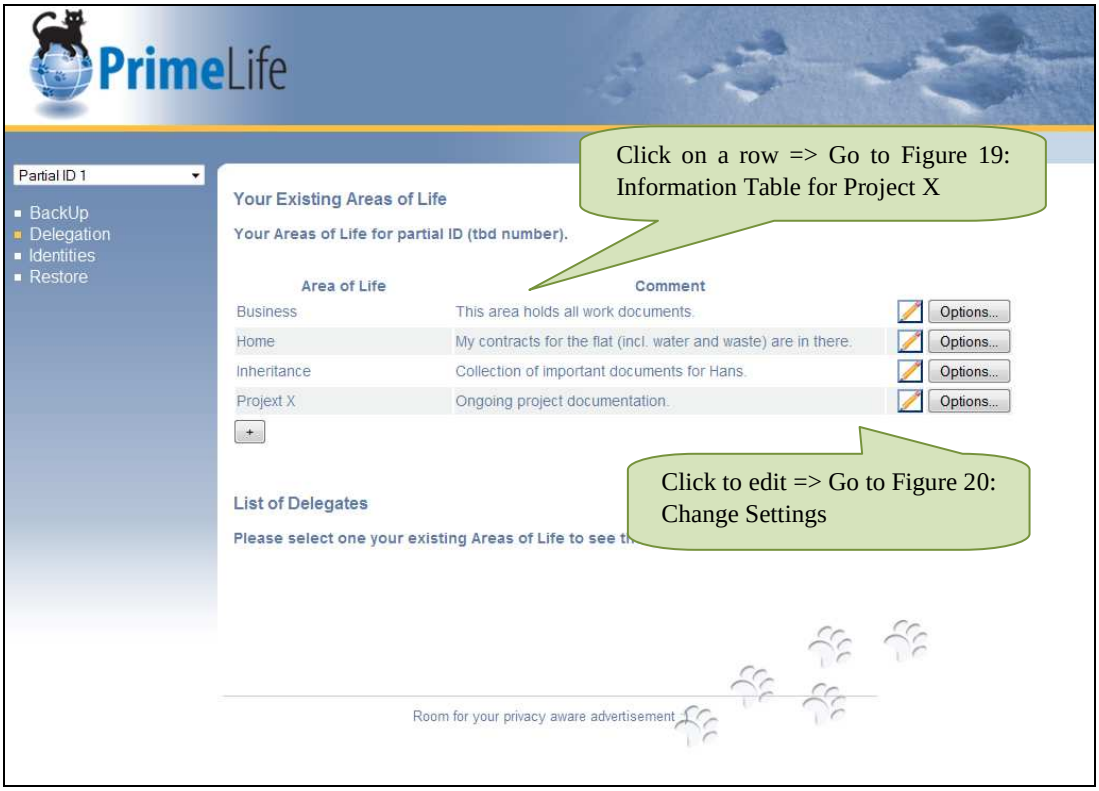

Figure 18: Delegation – Main

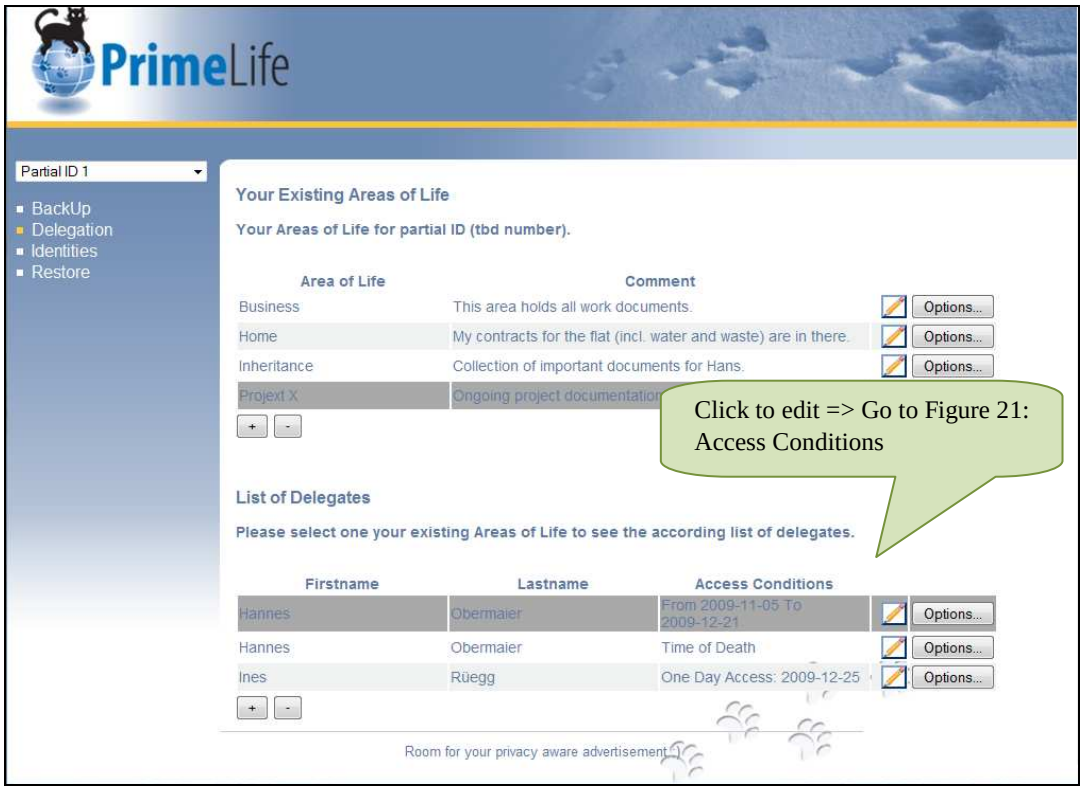

Figure 19: Information Table for Project X

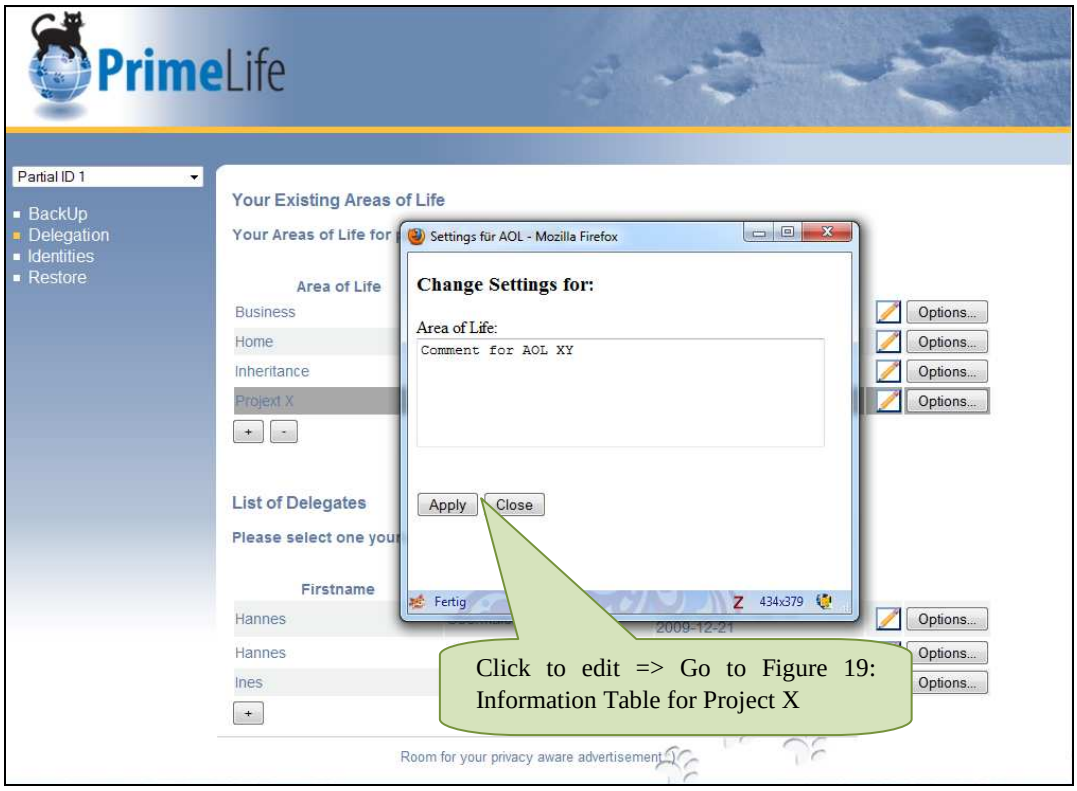

Figure 20: Change Settings

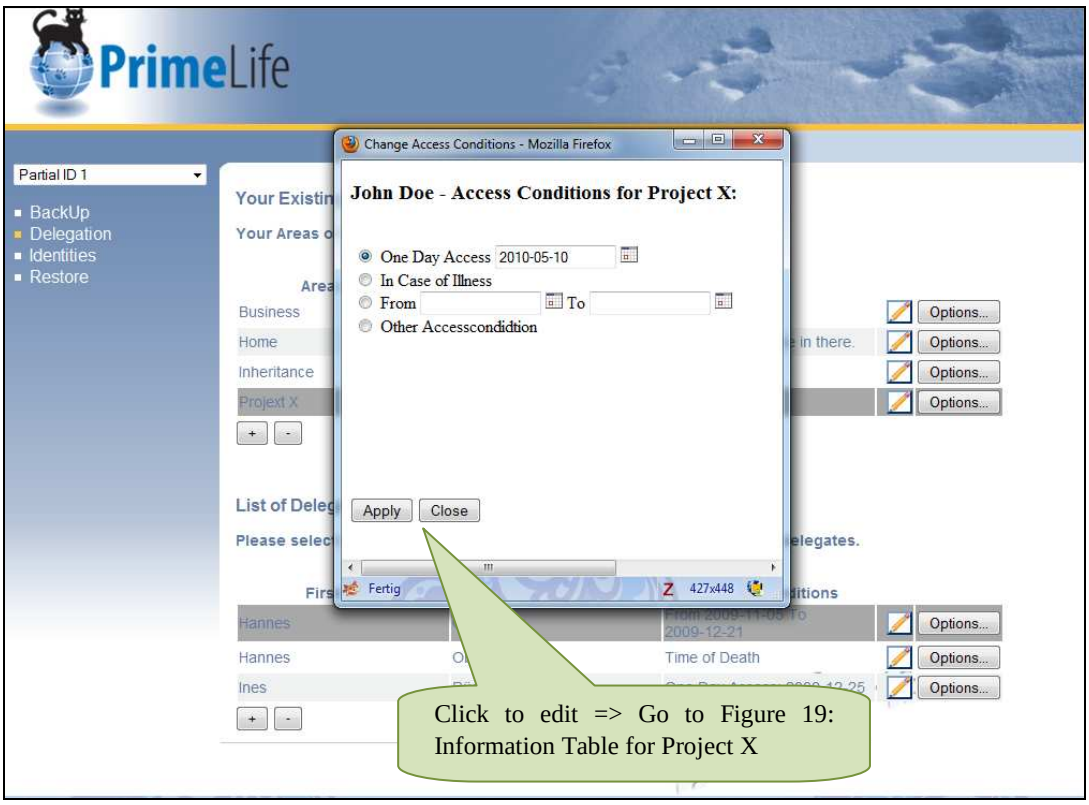

Figure 21: Access Conditions

2.2.3.3 Identities

The Identities-screen provides a list of delegates. Furthermore, the user has the possibility to get the assigned AoLs for each delegate.

We are aware that “Delegates” will be a better labeling. Anyhow, the navigation entry above is called “Delegation” and therefore we decided against “Delegates” as labeling to reduce cognitive load for end-users.

This screen is sort of opposite of the delegation tab – there the main focus lays on the AoLs of the user, in the identities screen the main focus is on the delegates.

Figure 22: Identities

2.2.3.4 Restore

The main function of Restore is to give the user the possibility to recover an AoL-backup.

The Restore screen also gives an overview about different backup versions of an AoL. So, it is not only possible to restore the last backup, but the user can restore any previous version of her backups.

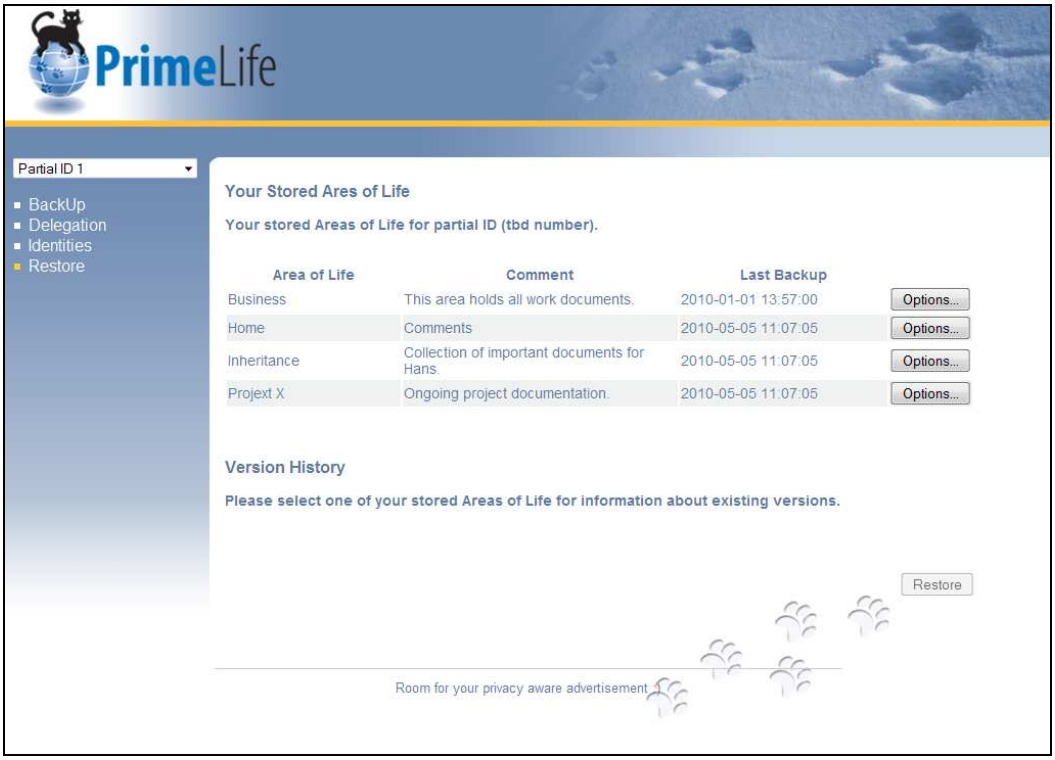

Figure 23: Restore

2.2.4 Part 2: Delegation

To get this view the user has to choose in the drop-down menu an entry "Delegation from '*delegates name*' ". This view provides the administration functionality for the delegations she got from another person.

The user has the possibility to look at all delegations she received from the selected person or to restore a backup originating from a delegation in case the access condition(s) for an AoL of that other person become(s) valid.

2.2.4.1 Delegation View: Delegations

One focus of attention during development of the delegation part was that a user should be able to decide whether she wants to accept or decline to be proxy[2] for another person within a given Area of Life. In addition, she should be able to reconsider her decision.

So, one problem was, to provide enough information that a delegate can understand which data an AoL contains and estimate the impact of accepting to become a delegate for this AoL. On the other side the information should not provide enough data to offend the privacy of the delegator until the access conditions to the archived data are met. Therefore we decided to implement the possibility of adding comments to an AoL. These comments shall be visible for the delegates in the column "content".

[2] Delegation is a process whereby a delegate (also called "proxy", "mandatory" or "agent") is authorized to act on behalf of a person concerned via a mandate of authority (or for short: mandate), cf. [5].

Figure 24: Delegations

2.2.4.2 Delegation View: Restore

This screen was not implemented in the HTML prototype.

In the future application this menu point shall provide the possibility to restore an AoL backup of when the parameter specified in the access conditions occurs.

Chapter 3

Privacy-Enhanced Event-Scheduling

This privacy-enhanced event-scheduling tool, called ‘Dudle’, allows users to schedule events. The underlying concept is as follows: some event organizer creates a poll where participants may enter their availabilities. The event can be scheduled depending on each participant’s availabilities. Additionally to event scheduling, the application can be used to make more general polls (e.g., about favourite movies, barbecue locations etc.) as well.

Usually within such polls, much personal information is disclosed (e.g., personal time schedules, personal interests movies, etc.). The advantage of Dudle consists of the possibility that single votes are hidden from other participants. Only the sums of all votes are displayed (one sum for every choice which has to be made). To avoid attackers, sending numerous dummy votes to modify the sums, every participant has to identify himself. To minimize the required level of trust in every party, cryptography is used for authentication and the individual votes are encrypted before sending. The whole protocol is out of scope of this deliverable, but the usage of cryptography has implications to the user interface (e.g., long cryptographic keys are needed; more computation is required, etc.). Furthermore, the problem that all participants identify themselves to the others is currently solved in a conventional manner (manual verification of cryptographic fingerprints), which is rather not user-friendly and should be addressed in further versions of the prototype.

For more detailed information about the display of the privacy relevant function of Dudle, please see Deliverable 4.1.3 [3].

Technical details of the underlying scheme were described in the PrimeLife Deliverable D2.3.1 as well as presented at PASSAT2009 [5]. An implementation is available at http://dudle.inf.tu-dresden.de; the source code is released under the terms of GPL and is available there as well.

Dudle is still under development; but it already reached a stable stage. We give an overview of the current version (revision number 462) at this point because the UI-design and software development were done within face-to-face collaboration of technical development and HCI.

3.1 Cooperative Design Process

The design process of ‘Dudle’ was settled in a surrounding where it was possible that the usability expert and the developer work in the same room. Such a setting allows live-exchange between

technical development and HCI. Daily ad-hoc expert review was given to the developer to highlight possible usability problems and give impulses how to improve them. In the face-to-face design process, direct input was given from an HCI-expert to the developer. A problem of the process was that the basics of the technical foundations have already been laid out. This led to the fact that some change requests from the HCI–side would have resulted in huge code refactoring and thus it had not been possible to realize the changes in the current version of Dudle.

From HCI point of view, the effectiveness of this cooperative design process can be improved by including HCI-considerations just from the beginning of the developing process. For future usage of a cooperative design process, we recommend that the cooperative design process should start in the beginning of the development, which makes the result more satisfying for both, the HCI-expert and the developer.

Another – location independent – method is the design process, which was used for the UI-development of the privacy-enhanced backup prototype.

3.2 User Interface Overview

In this section, we will give an overview of the current user interface of Dudle, cf., Figure 25.

3.2.1 Tab-Navigation

The tabs in the top navigation of the screen are arranged in groups, which belong thematically together. Therefore, it is easy for the user to navigate through the functions of the tool and work with them.

3.2.2 Poll Related Tabs

The poll view consists of two parts, the poll itself and the ‘Comments’-section (cf. Figure 25).

The upper part – the poll – is structured as a common table. This table contains all relevant information about the poll and also the possibility to take part in voting in the poll. The display of the results in the table differentiates between anonymous votes and non-anonymous votes. The results of an anonymous vote are displayed in blue, so the single selections are irreproducible.

When voting non-anonymously, the chosen options are visible for everyone. The tool provides two additional presentations of the results; on the one hand by symbols, on the other hand using color. These two representations give the user a first impression of which voting possibility is accepted and which one is less favored.

3.2.1 Administration Tabs

These tabs provide administration options for the current poll. Such options may concern the poll itself but also privacy-related topics.

When creating a new poll, privacy options are not activated; it is necessary to set them up for each new poll. These privacy options contain access control and invitation functionalities. Access control means that users need to login to be able to vote. Further, administrator password is required for changing privacy options. By inviting participants, the privacy-enhanced voting becomes possible. The poll-initiator has to activate the possibility for each invited person.

Options concerning the poll itself are the possibility to edit the columns after setting up the poll as well as deleting the poll.

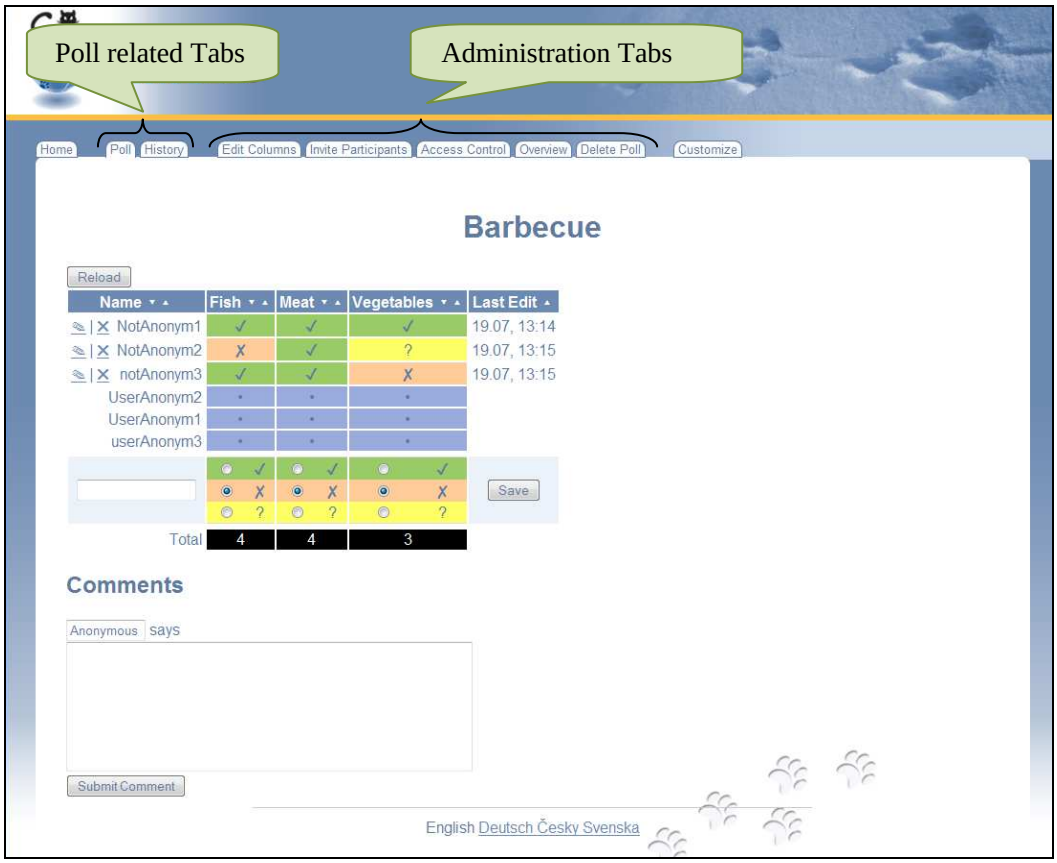

Figure 25: Dudle with non-anonymous and anonymous votes

Chapter 4

Credential Selection

A fundamental privacy design principle is data minimization, meaning that services or applications should be designed in accordance with the aim of collecting, processing or using no personal data at all or as little personal data as possible. Data minimization limits the communication partner's ability to profile users and is well acknowledged by most western privacy laws as a legal principle. It can in particular be derived from Art. 6 I (c), 6 I (e) of the EU Data Protection Directive 95/46/EC [2] and is, for instance, also explicitly required by the Section 3a of the German Federal Data protection Act [4].
Anonymous credentials in contrast to traditional credentials allow a user to selectively reveal only a subset of her attributes or to prove that she has a credential with specific properties without revealing the credential itself or any additional information. Additionally, the anonymous credential system Idemix , that has been developed by IBM, has the property that different credential shows are unlinkable, which can prevent the linking and profiling of different user sessions. In addition to Idemix used in PrimeLife, which IBM plans to contribute as Open Source, also Microsoft now integrates the U-Prove anonymous credential technology into Windows Communication Foundation and Windows CardSpace [1]. Hence, in the future, anonymous credentials will play an even more significant role not only in research but also in practice. For the successful deployment of anonymous credentials as a privacy-enhancing technology, its usability will be of key importance. In particular, users need to comprehend the data minimization property, which can be achieved by using anonymous credentials, so that they can fully appreciate their privacy features and have an increased interest in adopting this new technology. The design of intuitive and easily comprehensible user interfaces for anonymous credentials is however, a challenging task as anonymous credentials are rather complex technical concepts that are unfamiliar to most end users. Furthermore, no obvious or direct real-world analogy exists for them, on which the user interfaces can be based.
As the UIs, user tests and test results described below have been presented elsewhere, e.g.[8], the following text will be focused on a number of example UIs intended to illustrated the rationality of the design process itself.

4.1 The Credential Selection UIs Design Process

As both Idemix and CardSpace UIs are framed within the mental model of cards, we began by creating a selection mechanism consisting of the full source cards (e.g. driver licence) of the

anonymous credentials. The basic idea here is to let the user select which source cards will be used as to prove a given attribute (See Figure 26).

Figure 26: Full source cards

The result of the user tests with this UI basically showed that users did not understand the data minimising properties of the application at all and that they overestimated the amount of information being sent to the data recipient. In order to highlight that information was actually being selected, we applied a number of basic usability techniques such as greying out the information not being sent and showing a little scissor around the parts that were about to be selected (See Figure 27).

Figure 27: Card with the selected information highlighted

The result of the tests with the highlighted information UI was pretty much identical to the results of the tests with the full source cards. So, we discussed the issue with our colleagues in PrimeLife. Amongst the many ideas that were given to us, the one that came up most often was to use black lines to show what information was not being sent. The basic idea here was that users are used to the idea of blacking out information that is not to be shown. It is a well-known procedure that, for instance, can be seen in newspapers when somebody's identity is being concealed (See Figure 28).

Figure 28: Card with black lines applied to conceal information not being sent.

The results with the black lines UI performed approximately as bad as the previously tested UIs. Users still overestimated the amount of information being sent. One possible reason for this might be that even though the data is being concealed it is still being sent in the same fashion as somebody's picture is still being published even if the picture is distorted with a black line over the eyes. During the post user test interviews one frequent comment was that it would be much better to show only the information about to be sent, rather than showing and in some way also concealing information that is not being sent. In response to this notion, we created a UI showing only the information about to be sent but we also included an icon size image of the source card. This was done in order to show the users that the identity of the issuer of the information was also about to be sent (See Figure 29).

Figure 29: Card showing only the information about to be sent.

The results of this round of users test were marginally better than the previous ones. This shows that the use of the source cards as mental model very effectively makes the users overestimate the amount of data being sent even if the source card is only presented as an icon. This led us to abandon the full source card mental model and instead explore the attribute-based model. The basic idea behind this concept is basically not to refer to source cards at all but rather to refer to attributes and pieces of verified information. Thus, in essence the major difference lies in the description of the capabilities of the credential selection application and the task of the user.

Figure 30 below shows an example of a UI where the user is asked to select the verifier of attributes rather than the source card as in the previous examples.

Figure 30: Selection mechanism referring to the verifier of a given attribute.

The results of the tests with attribute-based UIs showed a much higher degree of correct responses than the tests with the card-based attribute UIs and, thus, it can now be said that the card-based mental model biases the users in an unfortunate way.

4.2 Conclusions

Within the PrimeLife project, we have conducted an extensive study on the users' comprehension and mental models of anonymous credential selection approaches at Karlstad University. In particular, we have investigated what effects the users' mental models have on their understanding of the data minimization property of anonymous credentials. For this research task, various UI mock-ups for anonymous credentials based on different UI paradigms and metaphors have been developed and tested in usability studies. In particular, we have investigated the suitability of a card-based metaphor for anonymous credentials as Microsoft CardSpace and other identity management technologies and initiatives use the information card metaphor. Different card designs/concepts were used to illustrate the selective disclosure property of anonymous credentials. Besides usability tests of the card-based anonymous credential selection paradigm, we also tested UIs based on an "Attribute-based" selection paradigm, where test users were told that they had imported validated attributes of information from trusted agencies.
The design process used to develop the credential selection UIs can be described as a receptive trial and error process. Trial and error in the sense that we have performed many short iterations and receptive in the sense that we have discussed the results with both test participants and colleagues from both academia and industry and incorporated their input as well as our tests results into the further development of the UIs. Given the difficult nature of the task to create UIs for a technology, which is inherently incomprehensible to users, this is probably the only feasible way.

Chapter 5

Conclusion

In this document, we presented three high-level prototypes. For each prototype, a different design process was used.
In general, we are able to say that each of the presented processes has pros and cons; the selection of the best process depends on various criteria, e.g. time, availability of participants, understanding of the users about the underlying concepts and several other factors.
The pro of the design process of the privacy-enhanced backup prototype was that no technical requirements were specified and no source-code was available by the time when designing the UI was started. This allowed us a very user-centred design focus during the development. The con of this method is that the outcome does not focus on the technical realization, so it may be hard for programmers to link the frontend with the middleware and backend of the application.
A similar approach was used during the development of the credential selection. Here, an iterative design process was chosen for the interaction between the development team and end-users. The pro of this design process is the continuing integration of users in the design-process. The con is that usability tests with end-users are very time-consuming.
The last process we presented was a design process, which took place between an HCI-expert and a developer. The upside of this process is that the designer gives usability input in each state of the development. Furthermore, this approach allows a constant intercommunion between different parts of the development team (in our case the HCI-expert and the developer). The downside is that this method works successfully only when development and design start coincidental.

Method used at	Pro	Con
Privacy-enhanced backup	• No technical specification / source code required	• No focus on technical realization
Credential Selection	• Integration of end-users	• Time-consuming
Dudle	• Constant intercommunion	• Development and design have to start coincidental

Table 1: Pros and Cons of used design methods

As mentioned before – all three processes have advantages and disadvantages and therefore it is not possible to say which one is the best. This depends on the background and circumstances the UI is developed in.

References

[1] Brands, S., Bussard, L., Claessens, J., Geuer-Pollmann, Ch., Pinsdorf, U., Credential Systems. In: in: The Future of Identity in the Information Society - Challenges and Opportunities, Chapter 4: High-Tech ID and Emerging Technologies, edited by Kai Rannenberg, Denis Royer, Andre Deuker/ Chapter 4 edited by Martin Meints and Mark Gasson, Springer, Heidelberg, 2009.

[2] Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data, Official Journal L No. 281, 23.11.1995.

[3] Fischer-Hübner S., Köffel C., Pettersson J.-S., et. al., PrimeLife Deliverable D4.3.1. HCI Pattern Collection – Version 2, Zurich, 2010.

[4] German Federal Data Protection Act, 1 January 2002, http://www.bdd.de/Download/bdsg_eng.pdf

[5] Hansen, M., Raguse, M., Storf, K., and Zwingelberg, H. 2010. Delegation for Privacy Management from Womb to Tomb – A European Perspective. In Privacy and Identity Management for Life, edited by Bezzi, M., Duquenoy, P., Fischer-Hübner, S., and Hansen, M. and Zhang, G. Springer Bosten 2010. pp 18-33.

[6] Kellermann, B. and Bohme, R. 2009. Privacy-Enhanced Event Scheduling. In Proceedings of the 2009 international Conference on Computational Science and Engineering - Volume 03 (August 29 - 31, 2009). CSE. IEEE Computer Society, Washington, DC, 52-59. DOI= http://dx.doi.org/10.1109/CSE.2009.270

[7] Lammi, J. 2009. http://uipatternfactory.com

[8] Malone, E. (Webmaster) 2008. Tool Tip Invitation: http://www.opendesignpatterns.com/index.php?title=Tool_Tip_Invitation

[9] Wästlund, E., Credential Selection, In Köffel, C., & Wolkerstorfer, P (Eds.) Low-Level Prototypes, 2009.

[10] Welie, M.v. 2008. http://www.welie.com/patterns