Enterprise Content Management – Archiving of Enterprise 2.0 data

Bachelor's-Thesis

For the degree of Bachelor of Science
in Informationmanagement

Author:
Nils Dammenhayn
Student ID: 209110706
E-mail: ndammenhayn@uni-koblenz.de
Date of dispatch: September 25th 2012

University of Koblenz-Landau
Faculty 4: Computer science
Institute for IS Research

Supervisors:
Prof. Dr. Petra Schubert
Prof. Dr. Susan Williams

Koblenz, September 2012

Erklärung

Ich versichere,
dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Mit der Einstellung dieser Arbeit in die Bibliothek bin ich einverstanden. Der Veröffentlichung dieser Arbeit im Internet stimme ich zu.

I declare that:

This thesis presents work carried out by myself and does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any university. To the best of my knowledge, it does not constitute any previous work published or written by another person except where due reference is made in the text.

Nils Dammenhayn

September 2012

Abstract

Regarding the rapidly growing amount of data produced every year and the increasing acceptance of Enterprise 2.0 enterprises have to care about the management of their data more and more. Content created and stored in an uncoordinated manner can lead to data-silos (Williams & Hardy 2011, p.57), which result in long search times, inaccessible data and in consequence monetary losses. The “expanding digital universe” forces enterprises to develop new archiving solutions and records management policies (Gantz et al. 2007, p.13). Enterprise Content Management (ECM) is the research field that deals with these challenges. It is placed in the scientific context of Enterprise Information Management.

This thesis aims to find out to what extent current Enterprise Content Management Systems (ECMS) support these new requirements, especially concerning the archiving of Enterprise 2.0 data. For this purpose, three scenarios were created to evaluate two different kinds of ECMS (one Open Source - and one proprietary system) chosen on the basis of a short market research.

The application of the scenarios reveals that the system vendors actually face the industry’s concerns: both tools provide functionality for the archiving of data arising from online collaboration and also business records management capabilities but the integration of those topics is not, or is only inconsistently solved. At this point new questions – such as, “Which data generated in an Enterprise 2.0 is worth being a record?” – arise and should be examined in future research.

Hinsichtlich der rapide anwachsenden Menge an jährlich produzierten Daten und der wachsenden Akzeptanz des Enterprise 2.0, müssen sich Unternehmen immer stärker mit dem Management ihrer Daten befassen. Inhalt, der unkoordiniert erstellt und abgelegt wird, kann zu Datensilos führen (Williams & Hardy 2011, S.57), welche lange Suchzeiten, unzugängliche Daten und in der Konsequenz monetäre Verluste hervorrufen können. Das „sich ausdehnende digitale Universum“ zwingt Unternehmen zur Entwicklung neuer Archivierungslösungen und Records Management Richtlinien (Gantz et al. 2007, S.13). Enterprise Content Management (ECM) ist das Untersuchungsfeld, welches sich mit diesen Anforderungen beschäftigt. Es ist im wissenschaftlichen Kontext des Enterprise Information Management angesiedelt.

Ziel dieser Bachelor-Arbeit ist es, herauszufinden in welchem Umfang aktuelle Enterprise Content Management Systeme (ECMS) diese neuen Anforderungen, vor allem die Archivierung von Daten aus dem Enterprise 2.0, unterstützen. Zu diesem Zweck wurden drei Szenari-

en erstellt, mit deren Hilfe zwei verschiedene Arten von ECMS (ein Open Source - und ein proprietäres System), ausgewählt auf Grundlage einer kurzen Marktübersicht, evaluiert werden sollen.

Die Anwendung der Szenarien zeigt, dass sich die Software Anbieter über die Probleme der Industrie im Klaren sind: beide Programme stellen Funktionen zur Archivierung von Daten aus online Teamarbeit sowie Möglichkeiten zum Records Management zur Verfügung. Aber die Integration beider Funktionalitäten ist nicht oder nur unvollständig gelöst. An dieser Stelle werden neue Fragen – wie z.B. „Welche im Enterprise 2.0 anfallenden Daten besitzen die Wichtigkeit, als „Business Record“ gespeichert zu werden?“ – aufgeworfen und müssen in zukünftiger Forschung betrachtet werden.

Keywords:

Enterprise Information Management, Enterprise Content Management, Enterprise 2.0, Archiving, Records Management, Tool-Integration

Table of Content

List of Figures

List of Tables

Glossary

CRM	–	Customer Relationship Management
E2.0	–	Enterprise 2.0
ECM	–	Enterprise Content Management
ECMS	–	Enterprise Content Management System
EIM	–	Enterprise Information Management
ERP	–	Enterprise Resource Planning
ISR	–	Information Systems Research

1 Introduction

This chapter provides an overview of the researcher's motivation and the aims and objectives of this thesis.

1.1 Motivation

According to a study by Gantz & Reinsel the "digital universe" will be 44-times as large in the year 2020, than it was in 2009 (Gantz & Reinsel 2010, p.1). In a decade where the amount of data being produced is growing that rapidly, its management becomes more and more important. As the acceptance of Enterprise 2.0 and collaborative work become more essential, enterprises need to support their employees in creating and sharing data (Fowell 2002, p.507/508).

However, this fact also leads to problems such as information redundancy, duplicated work and undefined responsibilities (Fowell 2002, p.508). Despite its diversity in emergence and presentation the key challenge is that increasing amounts of data have to be collected, stored and made accessible to the right persons. The keyword in handling this challenge is Enterprise Content Management (ECM).

This term encompasses the coordination of an enterprises' produced data, its presentation and thereby its contribution to an enterprises' success (Fowell 2002, p.509).

In many cases enterprises do not even know which data is available to them (Gantz & Reinsel 2010, p.7). The produced content that is stored may be uncoordinated and inaccessible and lead to unstructured data-silos (Williams & Hardy 2011, p.57). The use of ECM-tools can prevent these incidents.

Another important aspect of ECM is the integration of data, produced in diverse enterprise applications to a central platform (Eggert & Gronau 2008, p.2). Alongside this comes the topic of information compliance: the maintaining of business records including e-mails or facebook messages and twitter posts as well as providing guidelines for the use of social media in enterprises (Williams 2011, p.16/17). These responsibilities, which are growing through the use of social-media-platforms, constitute big challenges for organisations (Hardy & Williams 2010, p.235).

That requires a well-thought-out archiving system. This marks the entry point for this thesis. While archiving in general has been a topic long before computers existed, in the form of just storing documents for the future, e.g. in libraries, the internet and especially its Web 2.0 or in this case Enterprise 2.0 components raise new challenges for enterprises like the creation of new archiving methods or records management policies (Gantz

et al. 2007, p.13). In the 1980s archiving was not really more than scanning documents and pictures, but over the years topics like cloud and mobile archiving or database archiving have arisen (Kampffmeyer 2011, p.3-8) creating additional complexities.

Another one of these is web-content archiving. Not only do organisations have to know about previous information on their websites for their own interests in business information (Kampffmeyer 2011, p.5), it is also required as a matter of compliance.

The previous points and the fact, that archiving of Web 2.0 content is not discussed deeply in literature show the necessity of research in the field of enterprise content management in general and archiving specifically.

1.2 Aim

This thesis aims to provide a detailed investigation into the archiving of unstructured web-content in a dynamic E2.0 environment. This investigation adopts a constructivist practical approach with a tight connection to real business problems. The outcome will be scenarios to evaluate ECM-Systems regarding their functionality in the field of archiving of web-content and a review of how far two current systems support this concern.

Due to its extremely dynamic character and the participation of many people the archiving of online team rooms, web-pages or wikis and blogs raises the following two research questions:

1. How is the archiving of unstructured web-content integrated into ECM-tools (on a functional level)?
2. Which ways for retrieving archived web-content are provided?

To reach this aim and answer the research questions a combination of research methods including action research, prototyping and an analytic-deductive analysis will be used to evaluate two selected tools: Microsoft SharePoint and the Open Source solution Alfresco by Alfresco Software Ltd. The reasons for the choice of these specific tools will be discussed in chapter 2.2.2. Another interesting question for the discussion of the findings is to examine if there are different solutions from Open Source and commercial providers or if there is already a “standard” solution.

2 Research Methods and Design

At first one or more research methods that assist in answering the research questions have to be chosen. In this chapter the scientific paradigm that is followed in this research project is outlined and the research design is presented.

2.1 Paradigms and Research Methods

This chapter provides an overview of research paradigms in Information Systems Research (ISR) and their criticisms. Subsequently one or more research methods, underlying the paradigm mentioned before, are chosen and justified.

2.1.1 Introduction

Over the last decades ISR is going through huge discussions about adequate methodologies (Kelle 2007, p.14). While the history of qualitative and quantitative research is illustrated very detailed by Kelle (Kelle 2007, p.25-56), the following chapters first of all provide a differentiation of scientific research methods in behaviouristic and constructivist research, that was amongst others described by Österle et al. (Österle et al. 2010, p.1-2). Another possibility for the grouping of research methods are qualitative and quantitative methods, respectively the use of qualitative or quantitative data.

2.1.2 Constructivist vs. behaviouristic Research

The constructivist approach has always been followed in German-speaking and Scandinavian “Wirtschaftsinformatik”. It primarily focuses on the improving of information systems’ structure (Österle et al. 2010, p.1). So it is a practical procedure that’s results can be transferred to real business problems (Österle et al. 2010, p.1).

The especially US-American correspondent to the German term “Wirtschaftsinformatik” is the Information Systems Research. The leading paradigm is behaviourism which describes the monitoring of information systems’ characteristics and their users’ actions (Österle et al. 2010, p.1).

This second approach is criticised as lacking practical use. According to Österle et al. this assumption is supported by the fact that graduates in information systems research are less often employed in economy (Österle et al. 2010, p.1). Constructivist research however is said to be less scientifically strict, serious or ambitious (Zelewski 2008, p.175).

2.1.3 Qualitative vs. quantitative Research

The terms qualitative and quantitative can be used to declare on one hand different methods of data analysis, on the other hand the structure of produced data (Witt 2001, p.2-4).

Quantitative data describes countable data for example from measures, questionnaires or tests. These data exist in an abstract form which means its interpretation has to be done by the user (Witt 2001, p.2). Two main aspects of quantitative data analysis are the sample's representativeness and the data's degree of standardisation (Witt 2001, p.3). In conclusion the researcher's aim is to create comparable data. Qualitative data usually contains its meaning in itself like texts, pictures or video. So the process of how the data was created is not as important as it is in quantitative research, but for full deduction its arising also has to be considered (Witt 2001, p.1).

2.1.4 Selection of Methods

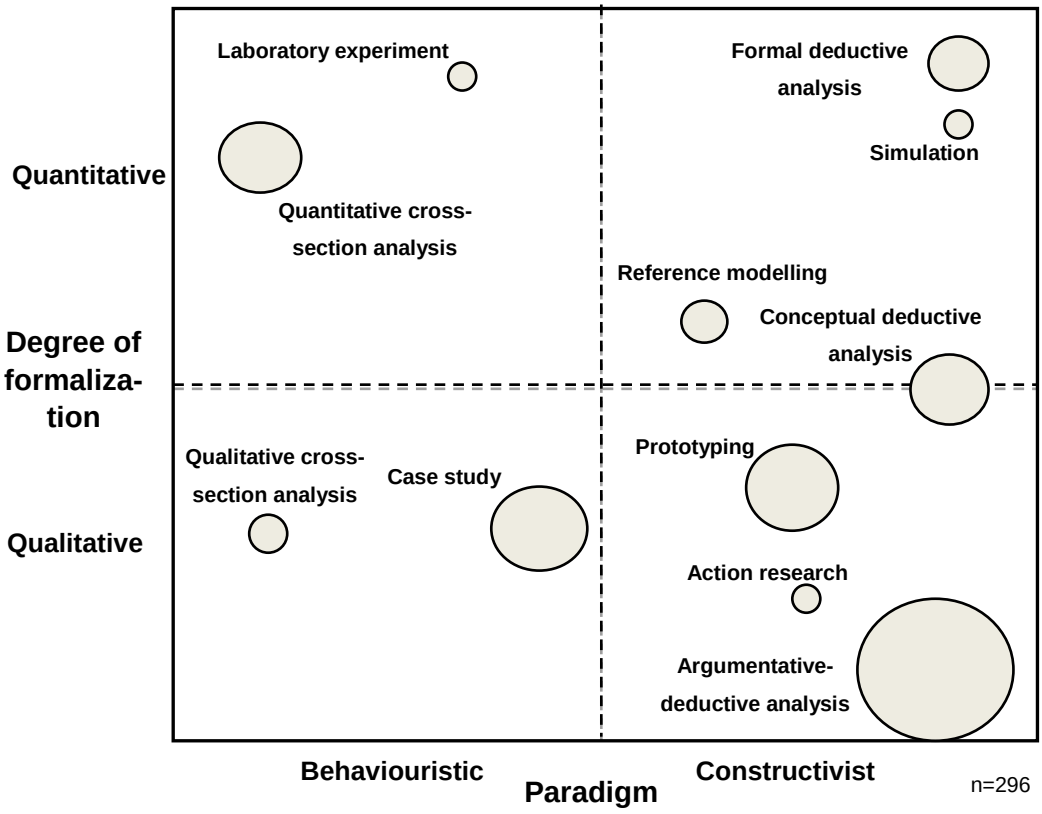


Figure 2.1: Empirical Profile of Research Methods in Information Systems Research (adapted from Wilde & Hess 2007, p.284)

Figure 2.1 shows a matrix representing the different research paradigms on one axis and data analysis methods on the other. The single research methods are weighted on how

often they were used from 1996 to 2006 for the “Wirtschaftsinformatik”-Journal which is displayed by the circle’s diameter (Wilde & Hess 2007, p.283). One can argue that for example the argumentative-deductive analysis is a generic term for many research methods – because in some way all research is argumentative – and appears as big for this reason, but the weight of the different methods is not the point of interest here. The reason for its mentioning here is that it provides a good overview of existing research methods placed into a research paradigm and type of data analysis, regardless of their weight.

As the issues raised in chapter 1.2 are of a highly practical nature this thesis primarily follows a constructivist approach. The focus on socio-technical tools causes the step towards qualitative data. According to these points this thesis is placed in the lower right corner of the matrix. As it is not possible to clearly state one research method here, different methods will be used more or less intense.

An important one in this project is action research. A widely-used definition is from Rapoport: “Action research aims to contribute both to the practical concerns of people in an immediate problematic situation and to the goals of social science by joint collaboration within a mutually acceptable ethical framework” (Rapoport, 1970, p.499). The mentioned “practical concerns” shall be pictured through scenarios. Scenarios, also called use-cases are used to describe how a persona (an imaginary user) behaves in a given situation or at a certain task (Courage & Baxter 2005, p.52).

2.2 Research Design

The following figure displays the thesis' research design. It consists of five phases that are introduced in the following.

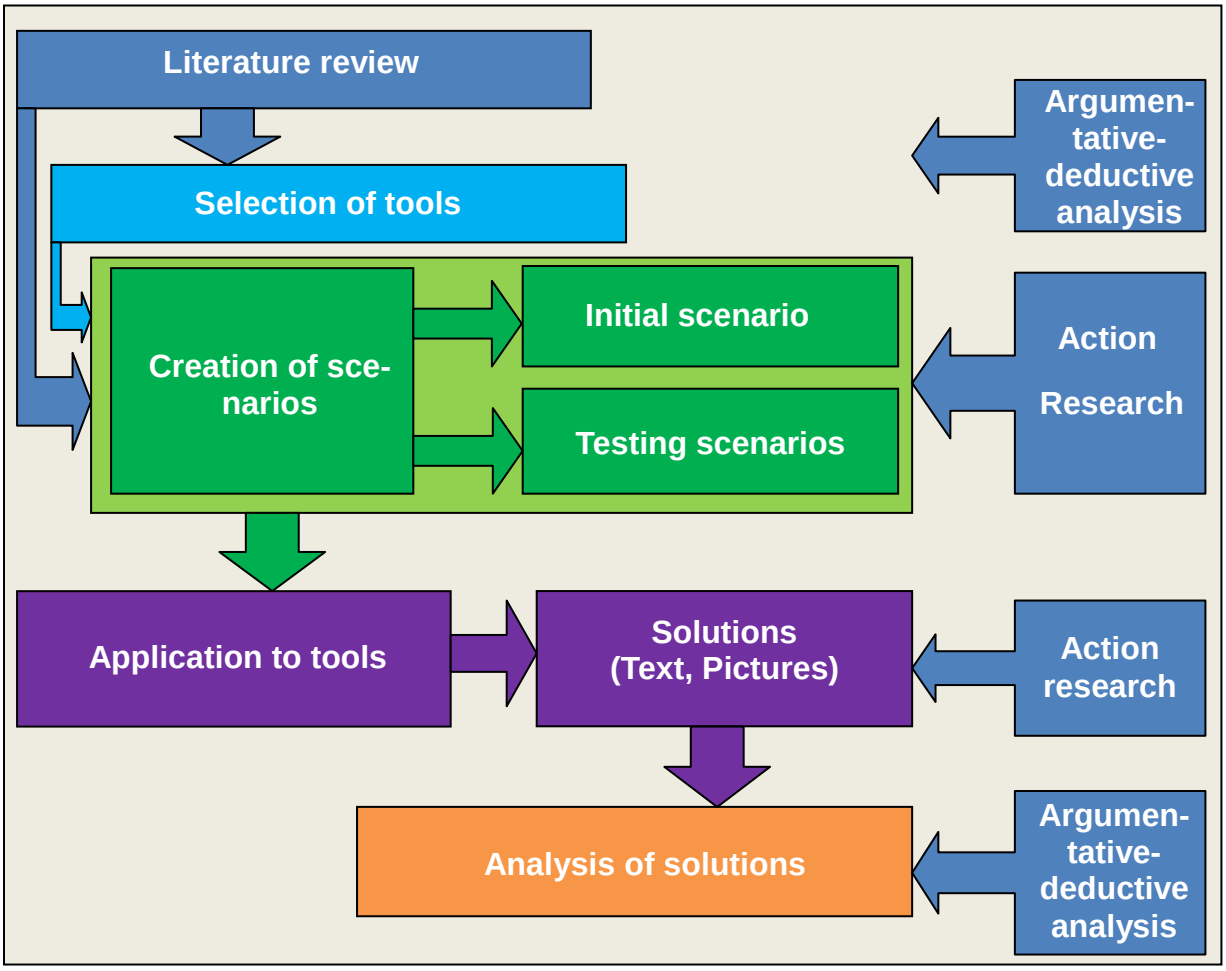


Figure 2.2: Research Design (own source)

2.2.1 Literature Review

This research begins with a literature review. At first the basics of Enterprise Content Management and its scientific context are introduced and then further specific research focussing on archiving is done (*see Chapter 3*). It points out that archiving in the meaning of "traditional archiving", for example the preservation of normal documents, seems to be a well-known topic in business and literature but the new kinds of archiving, already mentioned before in chapter 1.1, like web-content- or database archiving raise new requirements for enterprises (Kampffmeyer 2011, p.4).

2.2.2 Selection of Tools

The issue, either to make a deep analysis of a few tools or undertaking a complete market analysis provided a decision point. As this research aims to understand how archiving is achieved and to provide practical guidelines for archiving of E2.0 in depth data is needed. This is obtained through deep knowledge of the tools which cannot be built by a broad analysis. In the end two tools with different characteristics have been chosen: SharePoint and Alfresco. The market for ECM contains by now numerous tool vendors as *Figure 2.3* shows. How these tools were selected from this large field is provided below.

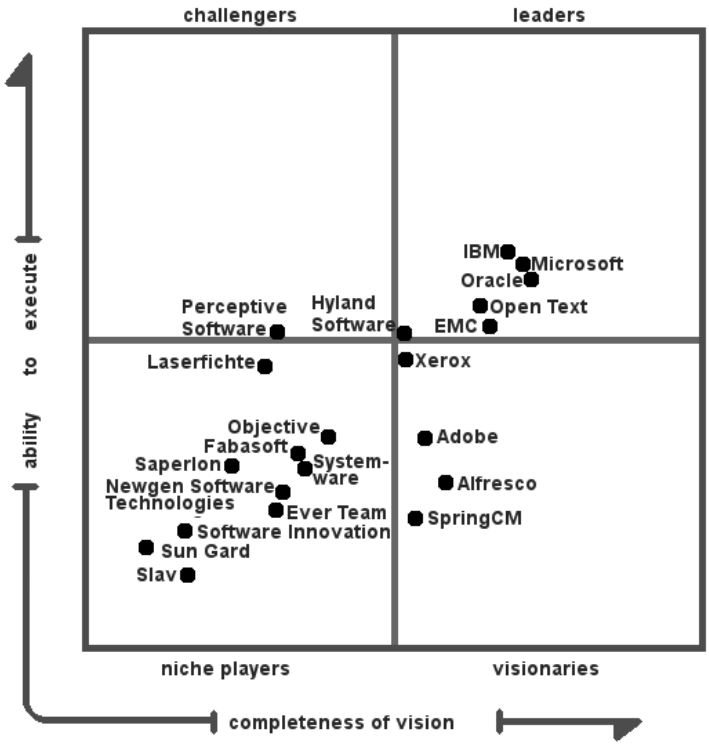


Figure 2.3: Magic Quadrant of ECM (adapted from Gartner 2011, p.3)

Due to the large amount of tools (see *Figure 2.3*) the key objective of the tool selection process is to identify outstanding ones.

Thus, the main criterion for selecting SharePoint is its widespread and numerous use in business which results in a high market share (Ovum 2010, p.5). According to Ovum this tool can almost be seen as a de facto standard in the ECM-world (Ovum 2011, p.12). Another reason for Microsoft's success in the ECM market can be seen in its well-integrated Office solutions that support employees in their everyday work (Ovum 2011, p.12). So this tool provides a good look into a system that is widely used for its ECM-functionality.

The Open Source solution Alfresco seems to be the opposite of a tool with high market share (*see Figure 2.4*), however Open Source tools are gaining more and more access into the world of business software (Ovum 2012, p.32). Especially the low costs and easier license models, which enable the development by independent developers, bring big advantages for most notably small enterprises (Ovum 2012, p.32). While in 2010 Ovum stated that Open Source solutions in general do not have the wide functionality of proprietary systems (Ovum 2010, p.4), in 2012 a trend to more extensive open solutions is detected (Ovum 2012, p.32). Thus, Alfresco was selected as an example of a different type of ECM solution that is gaining popularity in the market. Despite of its small market share, according to Ovum Alfresco is a challenger on the ECM market (*see Figure 2.4*).

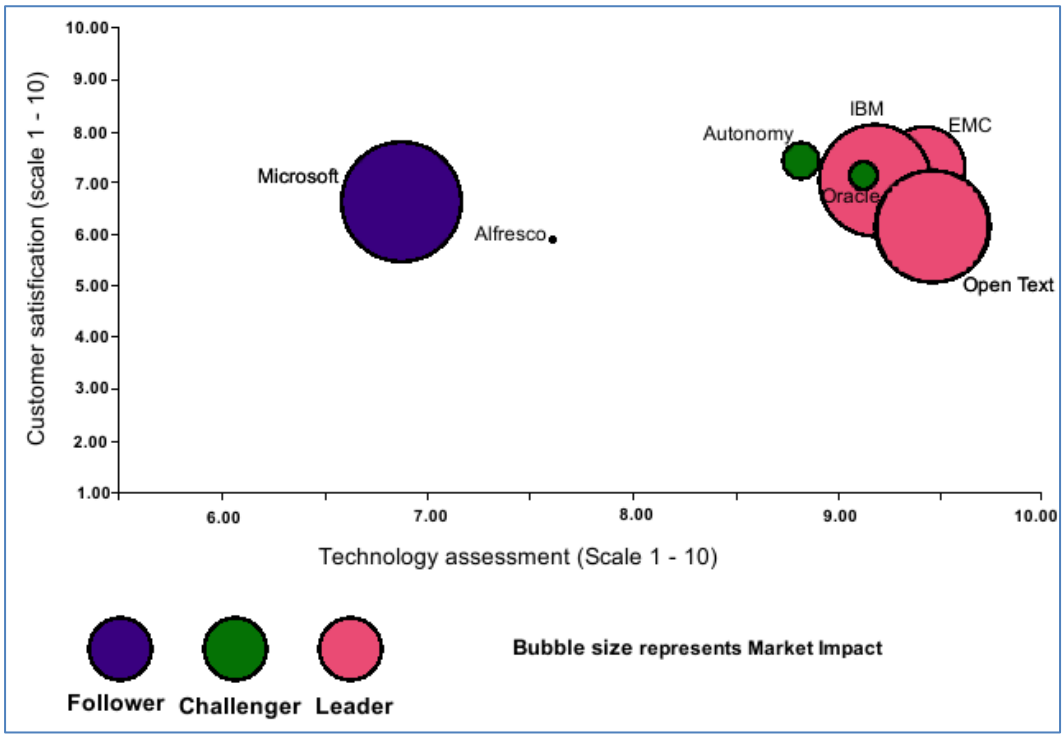


Figure 2.4: Decision Matrix for Content Management (adapted from Ovum 2012, p.72)

2.2.3 Creation of Scenarios

In order to examine and evaluate the selected tools a series of scenarios have to be created.

The first type of scenario is the *initial scenario*. In this document the starting position for the evaluation is set. It contains a description of the testing situation with a specification of the amount and type of documents, created and used by different users in each system. This scenario provides the basis for setting up and adding the same content to both of the systems under evaluation.

The next scenarios are the *testing scenarios*. These documents provide use-cases that shall lead to possible solutions for the problems raised in the research questions. Each scenario is proposed to give an answer to one question.

2.2.4 Application to Tools

This step is the actual evaluation phase and the primary data gathering takes place. Starting with the input from the initial scenario the tools are tested for their archiving and retaining capabilities using the testing scenarios. The testing processes' outcome is documented via screenshots which can be found in the Appendix.

2.2.5 Analysis of Solutions

After the scenarios have been applied to the tools the different solutions will be analysed and compared regarding their scope of functionality, usability and transparency. At last the outcome will be discussed and an outlook on further research that arises from this thesis will be given.

3 Literature Review

Chapter 3 provides the current state of research in the field of ECM.

3.1 Enterprise Information Management

Following a top-down approach at first ECM will be placed in a wider scientific context: Enterprise Information Management. In order to have a clear way of identifying where the topic “archiving” can be placed, after comparing two definitions of EIM the 8C-Model for collaborative technologies from Prof. Dr. Susan Williams is introduced.

3.1.1 Definition

A comprehensive definition of Enterprise Information Management is given by the German business consultancy “Project Consult” owned by U. Kampffmeyer: it stands for the complete management of enterprise information, independent of place, user, author, system, application, format, device and time. It combines functional approaches of Enterprise Content Management, Business Process Management, Enterprise Search, Business Intelligence, Governance-Risk-Compliance Management Infrastructure, Data Warehousing and Information Lifecycle Management (Project Consult 2011). So EIM can be seen as an umbrella term that includes ECM as one part of it.

Gartner defines EIM as “an integrative discipline for structuring, describing and governing information assets, regardless of organizational and technological boundaries, enabling business insight” (Gartner (2) 2012).

Both definitions describe the wide range of EIM covering all enterprise information. While ECM focuses on a technical side EIM claims to manage all enterprise information transparently to reach the business objectives.

3.1.2 The 8C-Model for Enterprise Information Management

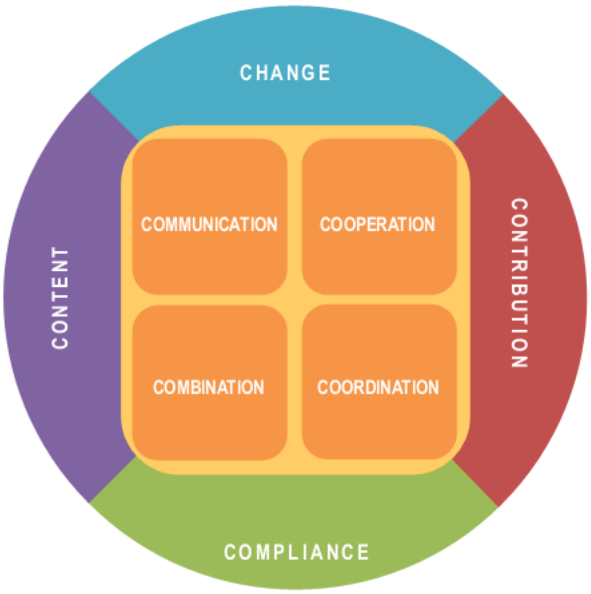


Figure 3.1: 8C-Model for Enterprise Information Management (adapted from Williams 2011, p.12)

The 8C-Model was developed by Prof. Dr. Susan Williams in 2011 as a framework for the analysis and evaluation of collaborative technologies (Williams 2011, p.11). The framework focuses on one implementation project of EIM technologies and helps to analyse the requirements for successful collaboration. It contains eight elements, which can be divided into the core and the outer ring with each four elements. Those shall be illustrated in the following chapters.

3.1.2.1 The core

The model's core consists of the elements *communication*, *cooperation*, *combination* and *coordination*. These terms represent the main activities that are necessary for successful E2.0-projects.

Communication describes all kinds of information exchange between human beings. This can be for example instant messaging, a telephone call, blog messages, etc. independent of time, place or the number of participants (Williams 2011, p.13).

Communication and *collaboration*, shown in one element, stand for any kind of work that is done by two or more people together. The difference between them is that collaboration is a more complex type of co-working with direct contact the other participants, while cooperation can be just the consolidation of two activities results into one, although they were finished completely independent (Williams 2011, p.14).

Coordination focuses on the orchestration of processes, actions and the resources they need. Especially workflow management is an issue in this core-element. Examplary applications are group calendars, awareness or room-reservation systems, always with focal point on the sequencing of activities (Williams 2011, p.14).

The last core element is *(content) combination*. Main task is the organisation of the produced data. This contains the improvement of data-recovering, -searching or –aggregation for example through tagging or RSS-feeds (Williams 2011, p.15).

3.1.2.2 The outer ring

The ring extends the model from a functional perspective to an enterprise level with look at the aspects *content*, *compliance*, *change* and *contribution*.

Content (management) describes the structuring of data, creation of metadata, rights management, as well as the management of information over its whole lifecycle (Williams 2011, p.16).

Compliance, as already mentioned before, is getting more and more important with the rising of E2.0. The preparation of social-media guidelines for employees to protect the enterprise from risks like the publishing of internal data is one task in this element of the ring. Another one is the compliance of governmental terms and law, for example which data has to be saved for how long and especially how this shall be done (Williams 2011, p.16/17).

Change management has to come along with every big IT-project. These projects can cause structural changes over the whole company and have to be guided. Employees who use the systems have to be made familiar with new ways of communicating with, for example, customers (Williams 2011, p.17).

Last but not least enterprises want to know about the *contribution* that collaborative technologies bring to their organisations. So this element focuses on monitoring of management ratios over the time to be able to calculate a possible return on investment, for example (Williams 2011, p.17).

In summary it can be said, that this thesis focuses on the outer ring, especially its element “content”. But that is not the only one: most content is generated in the core, but managed in the outer ring so another question is: How can data get from the core to the ring? In other words, how is content that is created in the core being effectively managed?

3.2 Enterprise Content Management

After the context of Enterprise Content Management has been identified as Enterprise Information Management and ECM as a part of it, the next step in the top-down approach is the definition of ECM and its components.

3.2.1 Definition

The term Enterprise Content Management was introduced by the AIIM about ten years ago and defined as “the strategies, methods and tools used to capture, manage, store, preserve, and deliver content and documents related to organizational processes. ECM tools and strategies allow the management of an organization's unstructured information, wherever that information exists” (AIIM 2012).

Another definition was made by Gartner: “Enterprise content management represents both a strategy to deal with all types of unstructured content and a set of software products for managing the entire content life cycle” (Gartner (1) 2012).

Both have in common that ECM is regarded from two perspectives: the first one is focused on methods and strategies to manage unstructured information, the second one on tools that support these efforts. But there is a difference between the definitions: Gartner talks about “the entire content life cycle”, which includes the creation of documents, while the AIIM definition only mentions capturing as first point. In fact, there are documents created in ECM-tools, e.g. blog entries or wiki pages, so that Gartner’s definition seems to be more complete.

3.2.2 Application Areas

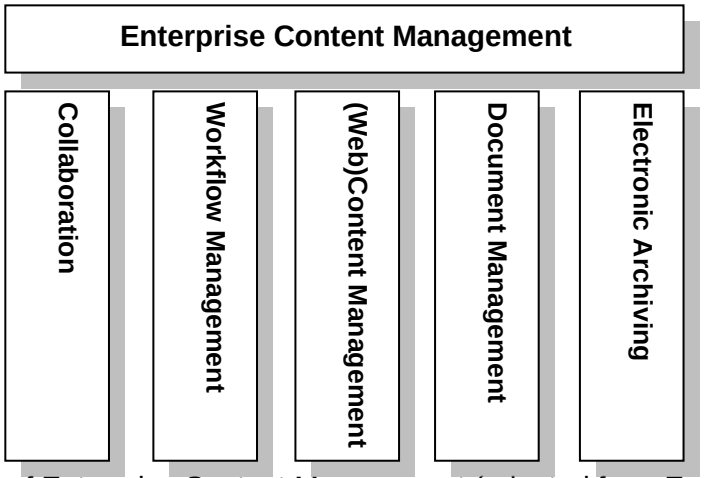


Figure 3.2: Overview of Enterprise Content Management (adapted from Eggert 2007, p.21, Kampffmeyer 2006, p.14)

The figure above displays a possible representation of the main application areas of ECM. While Eggert (2007) named “Knowledge management” in the original figure, Kampffmeyer (2006) chose the word “Collaboration”. This term is adapted here, because the field of collaboration seems to be a more independent application area, not only in ECM, but also in Enterprise Information Management, already mentioned in chapter 3.1.2 (Williams 2011, p.11-21). Knowledge management can be seen as a wider process that reaches over all other areas.

ECM is seen as an “integrative middleware” (Kampffmeyer 2006, p.10). While enterprises have ERP-, CRM- and other solutions as isolated platforms, the integration into one single platform is another task of ECM and brings diverse advantages: data is provided independently from its source, which leads to less redundancy, because it does not have to be shared over the systems (Kampffmeyer 2006, p.10).

3.3 Electronic Archiving

Archiving is a very important topic for enterprises: not only for compliance reasons, but for capturing the enterprise’s knowledge (Kampffmeyer 2011, p.9). This importance can be seen in the fact that there are even different standards stated to define how to capture business records: for example the ISO 15489 from the International Organization for Standardisation (ISO 15489-1: 2001) or the DOD 5015.2 Standard from the American Department of Defense (JITC 2007). The ISO 15489 defines records management as “[the] field of management responsible for the efficient and systematic control of the creation, receipt, maintenance, use and disposition of records, including the processes for capturing and maintaining evidence of and information about business activities and transactions in the form of records” (ISO 15489-1: 2001).

Olson tries to give an overview of enterprise content (see *Figure 3.3*) that contains different forms of digital data and also physical documents. But the socially arising content from wikis or blogs is not mentioned.

Kampffmeyer talks about six categories of new requirements according to archiving: database-archiving, universal archives, archiving of web-content and web-transactions, archiving of mobile applications and archiving in the cloud (Kampfmeyer 2011, p. 4-8). Partly overlapping with figure 3.3 his categories primarily focus on digital content arising in new technology. Especially interesting in this thesis context is the point “archiving of web-content” where he mentions that topics like how to archive wikis, short messages or social communities in general, are still not solved (Kampffmeyer 2011, p.6). This is exactly the point for this thesis to start.

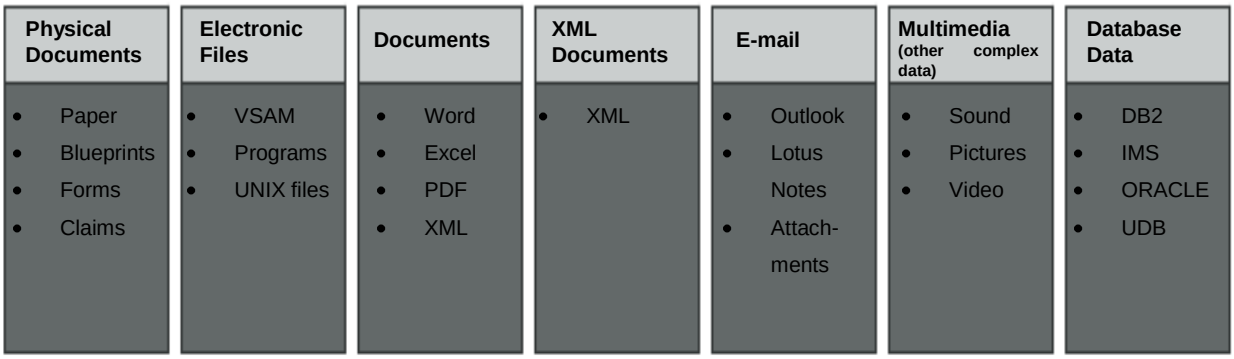


Figure 3.3: Categories of Enterprise Data (adapted from Olson 2009, p.6)

4 Scenarios

This chapter introduces the scenarios. The first one provides the background for the testing processes in the second and third scenario.

4.1 Initial Scenario

The table below states the starting point of the tool evaluation. As already mentioned in chapter 2.2.3 it defines members, type and amount of documents and the working environment with the general aim of the project and its background.

Project group & background	The research object is a fictive project group consisting of five members: a team leader and four associates. While two of them are working in Europe, the other members have their office in Australia. These people have the common aim to finish a programming task. To enable efficient collaboration despite of different time-zones and locations an ECMS is used to share knowledge.
Working environment	The project is organised in virtual communities or team rooms. These team rooms provide a restricted working environment for selected members with different roles. While the team leader has an administrative role, the normal members stay in the role “member”. The roles determine the level of access within the team room.
Functionality	The team rooms provide a wide range of functionality. All functionality relating to Enterprise2.0 is part of the investigation. The following list contains content creating functionality that could be in use: • Wikis: contain different kinds of user created data. That can be for example simple texts, spread sheets or pictures, created and put together by different users. Usually all pages have a version control. • Blogs: are filled with entries created by a single user. That can be texts like a meeting protocol or just information for other users. • Forums: provide a place for discussions or questions as an asynchronous channel inside a group of people. • Dashboards: aggregate information from the different sources inside a team room. This site gives awareness about who did

	what in the team. • Tagging: this component of E2.0 functionality gives users the possibility to describe the created content with keywords. Those keywords are displayed in lists or clouds to allow structuring and retrieving of information and data. • Shared documents: represent the traditional document management. User can upload files and share them with colleagues. • Shared bookmarks: provide functionality to simply link webpages for other users or themselves.
Current state	This scenario describes the picture of a typical project group collaborating via an ECMS. At the beginning of the investigation the project is in an advanced state. That means that various data have already been produced in the different applications provided by the particular tool. Thus, there are for example wiki- and blog-entries, shared documents and the members have discussed problems in forums.

Table 4.1: Initial Scenario (own source)

4.2 Testing Scenario 1: Project End

The first testing scenario aims to answer research questions 1 (*see Chapter 1.2*). It provides a situation that has to be solved using the tools.

Situation	In this scenario the members reached their aim to finish the project by completing their programming work. So the final code is written and uploaded and there are no more updates on any sites, documents or data. From this point on the team room is not changed anymore which means it is ready for archiving.
Goal	This is the entry point for the scenario. All the produced knowledge has to be stored in some other way than just leaving it where it is. The aim is to find one or more ways to get the data out of the team to store it.
Approach	The researcher tries to find functions or workflows provided by the tools to capture and manage the information produced during the project.

Table 4.2: Testing Scenario 1: Project End (own source)

4.3 Testing Scenario 2: Retrieving archived Information

Analogous to the previous chapter a scenario with a certain task is provided here, this time according to research question 2 (*see Chapter 1.2*).

Situation	This scenario is located half a year after the first one. A former team member has problems reinstalling his programming environment. He remembers a solution to his error was posted in the team room by somebody else half a year ago.
Goal	The aim is to find a way to retrieve information from an archived team room. In this case an answer from a discussion is searched.
Approach	This scenario bases on the outcome of testing scenario 1. According to the characteristics of the archiving solution, the researcher is looking for a way to retrieve certain information from this archive.

Table 4.3: Testing Scenario 2: Retrieving archived Information (own source)

5 Tool evaluation

The following chapter provides the tool evaluation process which is documented by screenshots.

5.1 Alfresco Community Edition

The first tool to be evaluated is the Open Source ECMS Alfresco Community Edition offered by Alfresco Software Ltd. The following table provides general information about the virtual servers' hard- and software that hosts the system.

Operating System	openSUSE 11.4 (64 bit)
Processors	2
RAM	2 GB
Hard-Disk	62 GB
Alfresco-Version	3.4 Community Edition

Table 5.1: Alfresco-Server Hard- and Software (own source)

The community edition offers collaboration tools for teams like wikis, blogs and forums and a document library to share files. Besides, there is a group calendar and the possibility to share links.

The next chapter gives a short general description of the applications and describes how they are used in the given scenario.

5.1.1 Application of Initial Scenario

The navigation through the different tools happens via a top menu bar (*see Figure 5.1*). Beneath, another horizontal menu bar provides the different functions for the particular tool. Referring to the wiki there is one button that links to the main page and one to create new wiki-pages (*see Figure 5.1*).

Figure 5.1: Top menu bar (own source)

Creating a team room, which is called site in Alfresco, is done by few clicks: from the home page after logging in (*see Appendix A-1*) one click on “Create Site” opens a pop-up window which asks for the name, URL, description and visibility (*see Figure 5.2*). Since the installation of Alfresco was done in German language it is not possible to change it anymore. The option “visibility” determines whether all users can join the site or if they have to be invited. This scenario provides a private team room with the name “Programming-Project Group”. A list of all team members can be found in Appendix A-2.

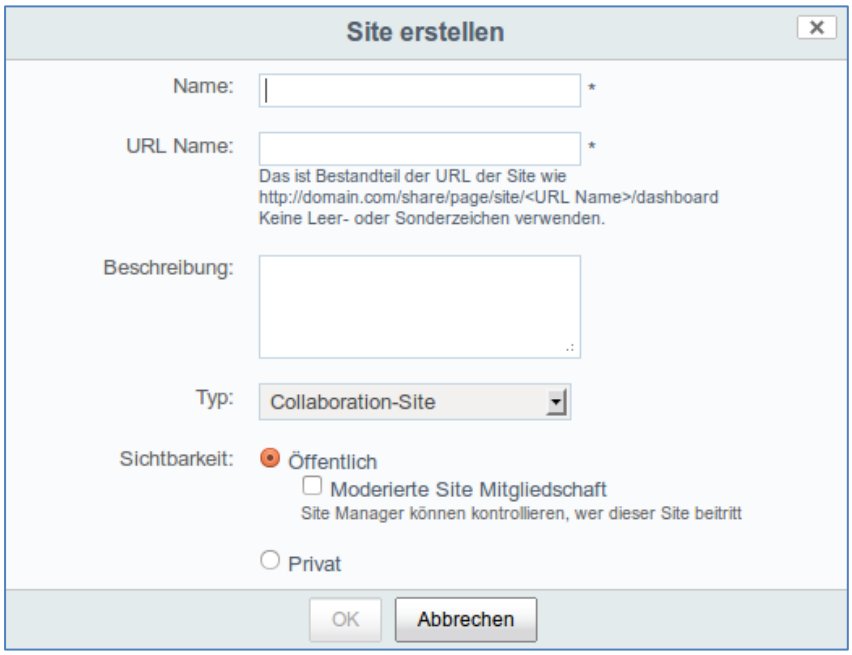


Figure 5.2: Alfresco Site creation (own source)

Alfresco’s home page for a team is a dashboard that summarizes all members’ latest activities (*see Appendix A-3*). This dashboard aggregates information from the whole site into one page. This happens by so called widgets. These small windows display other members’ actions in the specific tools. Figure 5.3 shows the activities widget.

All activities over a certain period are listed here with date, time, name and user. In this case the time period is seven days and can be determined by each user. The other widgets on the standard dashboard are for documents and links (*see Appendix A-3*).

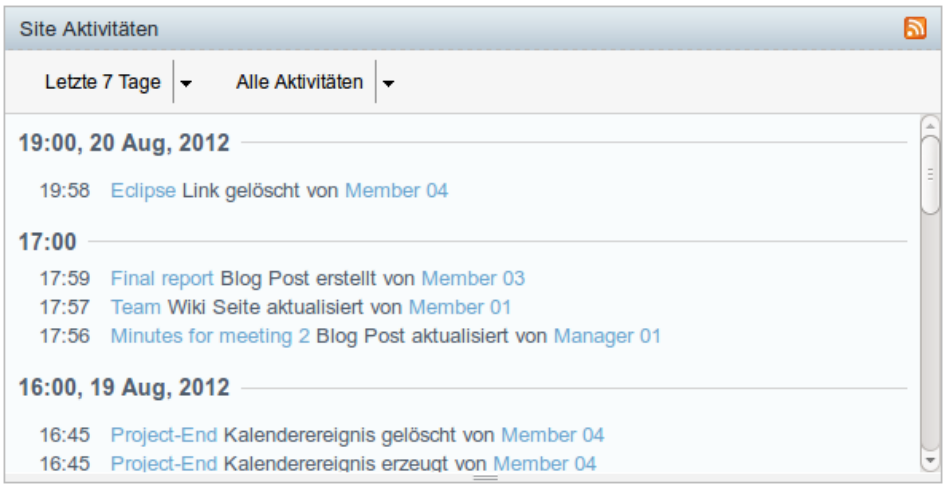


Figure 5.3: Site activities (own source)

The next function in use is the wiki. This tool enables team members to document their knowledge in a transparent way and make it possible for others to read and extend it. Appendix A-4 shows an overview of the wiki pages and the sidebar. In this sidebar users have the possibility to choose between lists of all, only lately changed or lately added pages. Besides, there is a tag-list to navigate to pages of a certain topic. Figure 5.4 shows how the wiki's main page looks like in this scenario. The users created different pages and linked them on the main page.

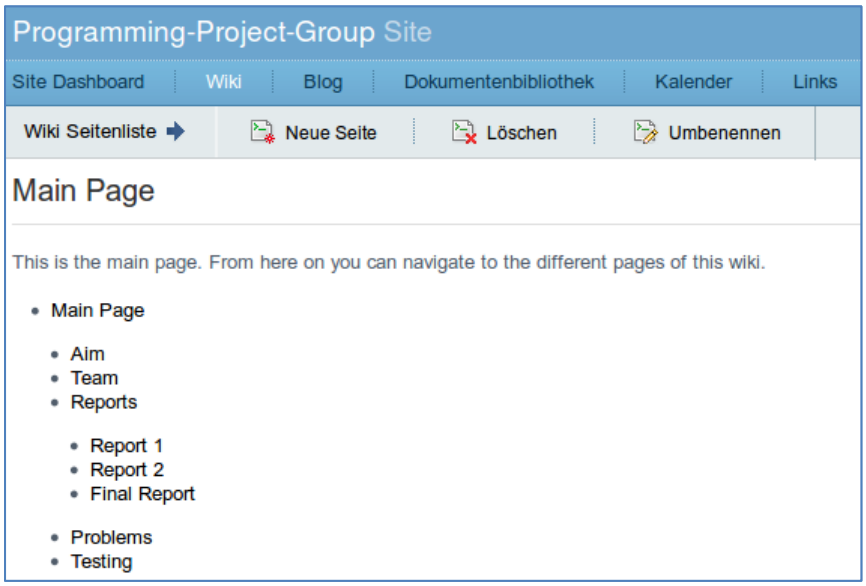


Figure 5.4: Exemplary wiki main page (own source)

Another typical web 2.0 application in this scenario is a blog. Different users can create single pages with information just as a memory for themselves or to share information

with colleagues, just as examples. While Appendix A-5 shows the overview page for this site's blog, Appendix A-6 displays a blog post that was commented by another user.

The document library provides an environment for users to upload all kinds of data and share them with others. Appendix A-7 shows the library's main page. The sidebar supplies navigation by tags, the folder structure that can be defined by the users or attributes like lately added or changed. In this scenario there are two folders with testing data in form of open-document text-files (*see Appendix A-8*), a pdf-file and one text-file created directly from Alfresco's library interface (*see Appendix A-9*). Besides there is a zip-folder that contains the final code (*see Appendix A-7*).

Next feature in use is a group calendar. The main page can be found in Appendix A-10. All site members are able to create calendar entries and add meta-data like place, time, description, tags and even a folder from the document library containing relevant documents for this entry. An exemplary entry with these meta-data is shown in figure 5.5.

Figure 5.5: Calendar entry with meta-data (own source)

The shared links functionality enables users to share useful links with their team. Again it is possible to add meta-data like a description and tags. The main page of the link-application can be found in Appendix A-11. Analogue to the previous tools the navigation is provided in the left sidebar, in this case only by tags or lately added links, a folder structure is not provided.

Last but not least the discussion application was tested. The home page gives an overview of all discussion topics (*see Appendix* A-*12*). In Appendix A-13 an example discussion with some answers to a question is shown.

The data list functionality is not tested in this scenario because there is no expression of this function in Microsoft SharePoint and a comparison is not possible.

Besides, each application except of the group calendar contains RSS-feeds. The icon can be found on the right side of the application's functional menu bar (*see Appendix A-4, A-5, A-7, A-11 and A-12*).

5.1.2 Application of Testing Scenario 1

As defined in chapter 4.2 the project group finished its work and the site is ready to be archived. To do so there are different ways which assume different levels of access into Alfresco.

At first some general facts about how Alfresco works have to be stated. There are two different user interfaces with separate login interfaces: Alfresco Share which is mainly used here and the Alfresco Explorer. While Share provides collaboration features the Explorer offers more administrative functions. In contrast to chapter 5.1.1 where only Alfresco Share was in use this chapter will need functionality from the Alfresco Explorer.

But at first back to Share: all content created in a site can be accessed through the application where it was created or through a so called repository. This repository can be accessed through the "Repository" button in Alfresco's top menu which is visible at all pages (*see Figure 5.6*).

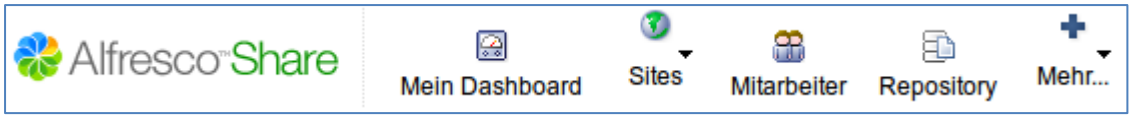


Figure 5.6: Alfresco's top menu (own source)

The repository's main page can be seen in Appendix A-14. The left sidebar again provides a navigation menu by folders which is used to navigate to "Sites". This folder shows all sites of the user logged in. By clicking at the folder with the site's name ("programmingprojectgroup") each application of the site shows up as a single folder (*see Appendix* A-*15*). Inside of these folders all content added to the site can be accessed and according to the user's role (member or manager) changed or deleted. Figure 5.7 shows the content of the folder "blog". All blog posts are shown as single files.

Figure 5.7: Content of the repository's blog folder (own source)

The special feature of this kind of accessing a site's content is that each file can be downloaded separately. The download-button can be seen when hovering over an item (*see Figure 5.7*). Pressing this button inside the "blog"-folder downloads the blog post as an html-file. Links, wiki-pages and discussions are also exported as html-files, while shared content from the document library stays in its original format. Group calendar entries can be downloaded as ics-files, which belong to the iCalendar-format. So they can be imported into other calendar applications.

This functionality provides a first approach to capture a site's content and store it outside of the team room that can be accomplished by every user. But this kind of manually archiving of single documents becomes inefficient very quickly if the volume of data increases. It only makes sense if the user exactly knows which blog post or file for example he wants to capture.

This problem can be solved by using Alfresco's WebDAV interface. WebDAV stands for "Web-based Distributed Authoring and Versioning" and describes an open standard for providing data on the internet. This interface can be mounted by common operating systems like Windows and diverse Linux distributions. Now the sites' content can be simply copied to the user's hard-disk or any other place. Appendix A-16 shows two pages of the WebDAV interface opened in Mozilla Firefox. The first one displays the test-site with each application's folder, the second one the content of the blog-application.

The third approach to collect a site's content requires the use of the Alfresco Explorer for the first time. Appendix A-17 shows the administration console's home page as an administrator. The following task can also be done as a normal user but those can only use places in the repository they own as target, while an administrator has access to the whole Alfresco repository. Clicking on "Company Home" in the left sidebar opens a folder structure to navigate similar to the repository mentioned above. Appendix A-18 displays the overview page of the testing site. Again each folder contains one application's con-

tent. Pressing the button “More actions” in the window’s upper right corner opens a drop-down menu with the entry “View details” (*see Figure 5.8*).

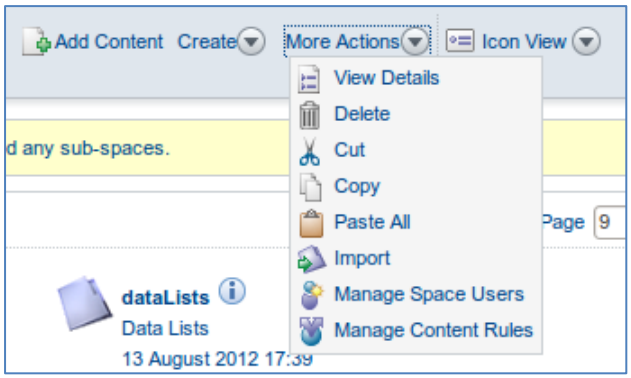


Figure 5.8: Drop-down menu to access a site’s details (own source)

Using this button opens a site’s detail-page (*see Appendix A-19*). On the right side of the screenshot a list of possible actions for this site is listed. The “Export”-button provides the possibility to export the whole site with all its content into an acp-file (Alfresco Content Package). This acp-file can be imported into other sites or opened by common-packaging-tools (*Annotation:* Opening this file in a packaging tool is not recommended as the files have no significant names). Appendix A-20 displays the configuration-window for exporting a site. Required fields are a package name and a destination. In this case the destination is the site itself, but another site or folder inside of the repository can be selected. The first hook defines if children (sub-sites) shall be exported. As there are no sub-sites in this scenario this hook is of no importance. If the second hook is not set, only sub-sites will be exported. The last hook “Run in background” gives the option to continue using Alfresco during the export. If not set, the next page does not load until the export completed. A click on “Ok” starts the export. Once finished the package appears on the site’s overview page under “Content Items” (*see Figure 5.9*). The button with the little red arrow provides a download-link for the package.

Figure 5.9: Site exported as Alfresco Content Package (own source)

The previously mentioned workflows are all manual tasks. Alfresco also offers functionality to define so called “rules” which indeed do not really export content, but are able to create a copy of every new data that is created in a site. Rules can be defined by any user but again only an administrator has the possibility to set any place on the platform as a target.

To configure these rules the site details-page has to be opened (*see Appendix A-19*). Under “Actions” on the right side there is the button “Manage content rules”. This button opens a wizard to create a variety of rules to apply on content created in a site. Each application can get own rules to ensure the export format fits the created content’s format. Appendix A-21 shows an exemplary rule that copies all created or updated blog entries into an html-file. This feature solves the problem of having data available even if the original site is deleted; it does not archive content but just copies it to another place.

The last approach was Alfresco’s records management capability. Since version 3.2 Alfresco Community Edition is certified to the already mentioned DOD 5015.2 Standard from the American Department of Defense (Alfresco Ltd. 2011). This functionality is carried out by a records management site which has to be activated (*see Figure 5.10*).

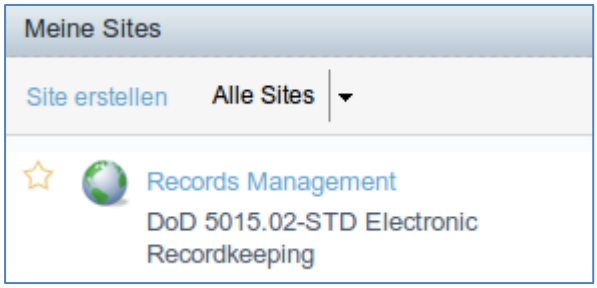


Figure 5.10: Records management site (own source)

Members of this site can access, declare and undeclared records according to their role inside the community. But this functionality is not integrated into group work sites. That means it is not possible to declare content directly from the team room as records. The single files at first have to be downloaded and then they can be uploaded into the records site. But even without the integration this functionality is suitable for archiving for example an exported acp-file in consideration of legal needs and the opportunity to retain it. Appendix A-22 shows the different record types Alfresco offers which also define the retention schedule. In this scenario the type “Web Record” is chosen. After uploading the specific file several meta-data according to its type like responsible user, publication date, URL etc. has to be added before the button to finally declare the file as a record

appears. Appendix A-23 shows the exported acp-file with the button to declare it. This file is now archived and retainable to the rules of the DOD 5015.2 Standard.

Another very practical feature of the records management site is the “Export all” button, shown in Appendix A-23. This button is available in all folders of the site and enables users to export and directly download all content inside of the specific folder and its subfolders into an Alfresco Content Package or a zip-file. The export dialog is shown in figure 5.11.

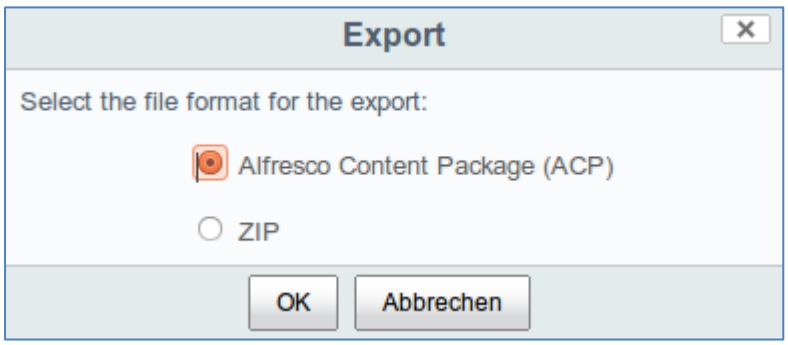


Figure 5.11: Dialogue for exporting records (own source)

The records management site does not support any functionality to alter records directly but with the appropriate rights on this site it is still possible to move or delete records. If this has to be prevented, for example in case of litigation, when a legal hold is put on certain files and any changes on them are forbidden, Alfresco provides a “freeze” function. If a certain file or folder is frozen it cannot be changed, deleted or moved until it is “unfrozen” again. Besides the file is copied to a special folder called “Holds” (*see Appendix A-24*) and all retention schedules are stopped for frozen records so it cannot be deleted by the system if its retention period expires.

Summing up Alfresco offers lots of different ways for all users to capture the content arising in collaboration sites, but most of them have to be triggered manually. Business records capabilities exist but are not integrated into group work sites, a fact that complicates their use.

5.1.3 Application of Testing Scenario 2

As mentioned in chapter 4.3 this scenario mainly bases on the outcome of the previous test.

The first approach assumes users to have access to an acp-file, generated previously. As there is no connection between the records management site and other team rooms an acp-file that was declared as a record has to be downloaded at first, before it can be

imported. The scenario defines that the old site is closed so a new site named “Archive” was created by the user. The import processes are similar to exporting a site so again the Alfresco Explorer has to be used. On the created archive-site’s detail-page the “Import”-Button is provided which opens a dialog where the source file is requested (*see Appendix A-25*). After the import process the user has access to the site’s overview page again. Appendix A-26 shows that the archived site is now displayed as a folder next to the other site applications. Following the scenario the user wants to find a topic in the discussion-forum where a link was posted. Opening the folder “projectprogramming-group” shows up all applications which used to be there. Inside the “discussion” folder all topics are listed as one file (*see Appendix A-27*). The problem here is that all topics have cryptic names so eventually some time has to be invested to search for the right one. Figure 5.12 shows an archived discussion. All other content of the site can be either downloaded or directly viewed in the browser this way.

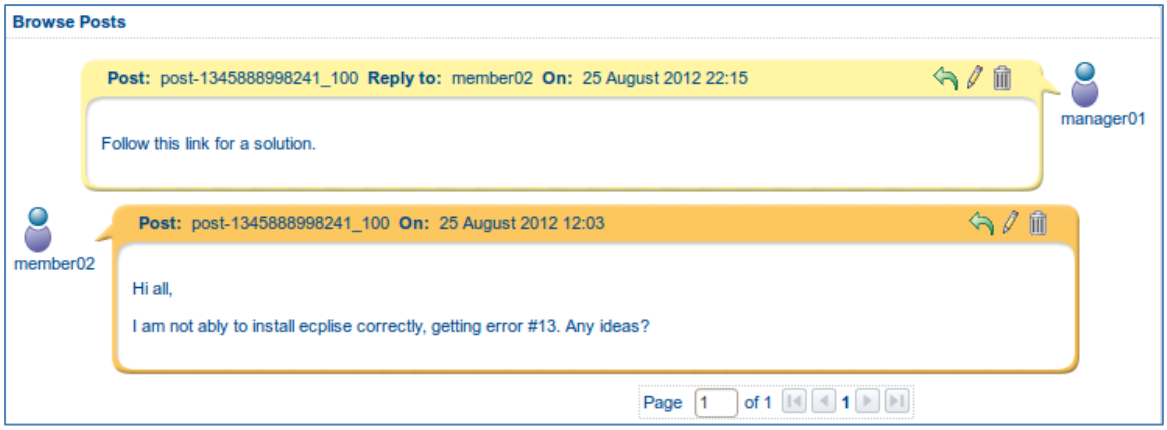


Figure 5.12: Archived discussion topic (own source)

5.2 SharePoint

The second tool is Microsoft’s SharePoint. Again a table with information about the installation environment is given.

Operating System	Windows 7 (64 bit)
Processors	2
RAM	4 GB
Hard-Disk	35 GB
SharePoint-Version	14.0

Table 5.2: SharePoint-Server Hard- and Software (own source)

Microsoft SharePoint offers a wide range of different tools that can be integrated into a team-room, also called site here. The testing site has the following applications in use: a blog, wikis, document- and link-sharing capabilities, a discussion board and a group calendar. Besides these there are applications like phone-call-memos and tasks activated by default but they are not tested because of no similar tool in Alfresco. Appendix B-1 shows the full list of all applications that can be added. Another interesting feature is the recycle bin which can be compared with the well-known Windows recycle bin. All content deleted is moved to this folder at first and not deleted before it is emptied.

As the single applications have been described with the first scenario's application in Alfresco, the next chapter only describes how they are used in SharePoint.

5.2.1 Application of Initial Scenario

The home-page after the log in can be seen in Appendix B-2. By default it displays the calendar showing the actual week, an announcements widget and one for links and new personal items. SharePoint provides the site's main applications mentioned in the previous chapter in a sidebar on the left (*see Appendix B-2*). The top menu can be seen in figure 5.13.

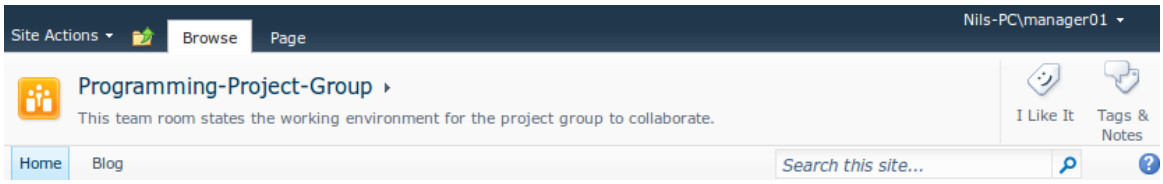


Figure 5.13: SharePoint's top menu (own source)

The button "Site Actions" opens a drop down menu with administrative functions like site settings and permissions and the possibility to create a new site or change the actual page's look. On the right side a search bar is provided and a button for marking the page as "liked" and a button to add tags to it. In the upper right corner a drop-down menu with user specific functions like editing the profile or signing out is placed.

The site-creation is quite simple too but offers more functionality. Site title and URL are required, a description is optional. The next configuration step is to select a site template (*see Appendix B-3*). These templates are sites with predefined applications for different purposes. This scenario uses a "Group Work Site" that is extended by a blog and a wiki. The next point defines if the access-permissions are adopted from the parent site or unique. The last three options declare in which navigation bars the new site shall appear. The "top links" in this scenario are "Home" and "Blog" and can be seen in figure 5.13.

The “Quick Launch” is displayed in Appendix B-2 as the left sidebar with links to sites, libraries (a SharePoint “library” can be compared to an application – the wiki functionality for example is one library), lists etc. The site’s member list can be found in Appendix B-4.

Again the first function tested is the wiki. The main page is displayed in Appendix B-5. The specific application’s features are available through the top menu. Figure 5.13 displays the “Browse”- and “Page”-buttons which enable the user to switch between one view to browse pages and one with application features. The feature-view is opened for wikis in Appendix B-5. It turned out that there is no button to create a new page but with a little trick it is very simple: the user has to create a link to the (till now not existing) page on the parent page. Clicking this link opens a dialogue which states that the page does not exist and asks if it shall be created (*see Figure 5.14*).

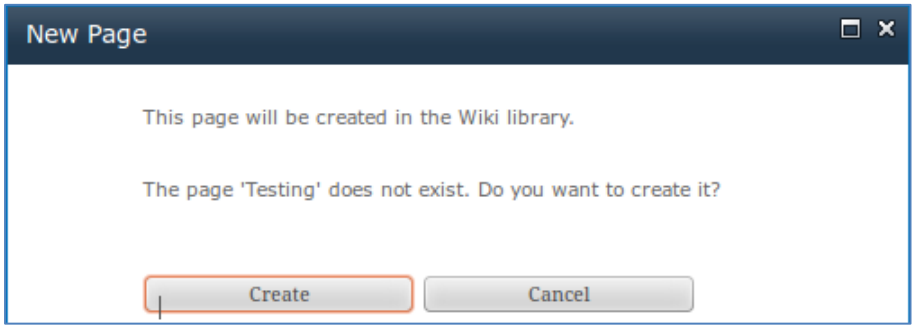


Figure 5.14: Wiki page creation (own source)

Next in row is the blog. In contrast to Alfresco a blog is not a standard component of team collaboration site in SharePoint and had to be added. The navigation through the blog functions differs from other applications as there is no “Page” button which offers the blog’s functionality. This time it is placed on the right side of the main page (*see Appendix B-6*). This inconsistency bases on the point, that a blog in SharePoint is not an application like a wiki but an own sub-site of the team-room. The biggest difference to Alfresco’s blog is the revision function. If required a blog post can be saved as draft. That means it is not displayed until some other user, in general a manager takes a look at the post and confirms it. Appendix B-7 shows the window for creating new blog posts. By clicking into the filed “Body” the top menu changes into a menu bar quite similar to Microsoft Word (*see Figure 5.15*).

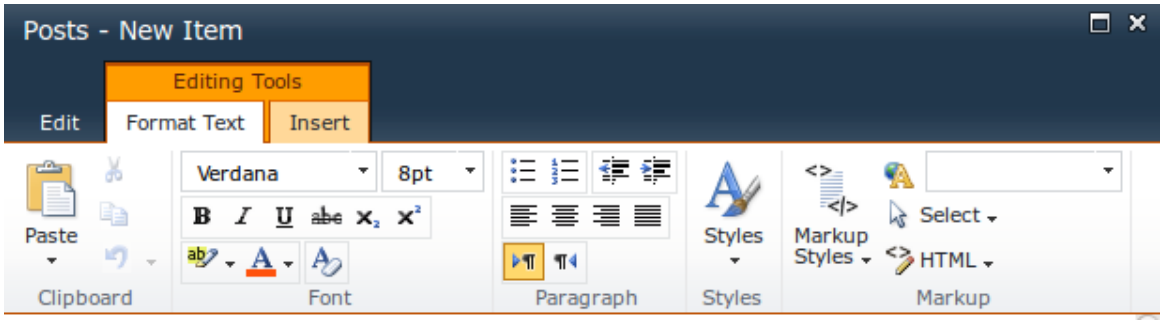


Figure 5.15: Blog post creation menu (own source)

In the Quick Launch bar on the left side of the main page an archive of the last months is arranged (*see Appendix B-6*). Besides the posts can be sorted into different categories at creation to access them here (*see Appendix B-7*).

The sharing of documents and shared editing is one of SharePoint's main capabilities. It discriminates between one library for shared documents in terms of shared editing and one for general data uploads and sharing of non-office files. Both are available via the Quick Launch sidebar (*see Appendix B-5*).

The main page for the "Shared Documents" application can be found in Appendix B-8. If at least Microsoft Word 2007 is installed on the user's computer a click on "New Document" opens a new file in Word. This brings the large advantage of being able to use all of Word's features directly in a document that is shared with others. Saving the file in Word creates a folder on the user's computer that synchronizes itself with SharePoint via the Microsoft Office Upload Centre application (*Annotation:* The application is automatically integrated into Windows' auto start without being prompted). But there is another disadvantage arising from this functionality: SharePoint is depending on Word. That means if Microsoft Word is not installed on the user's machine, he has no possibility to create or alter documents directly in SharePoint. In contrast to that Alfresco has at least a plain text editor with html- and xml-capabilities built in. In this case documents have to be uploaded via the button "Upload Document". Besides there are many other functions like a version history, different sharing capabilities which will be described later or workflow and publishing functionality (*see Appendix B-8*).

The "Documents" library provides the same top menu bars as the "Shared Documents" library. As the "Documents" menu can be found in Appendix B-8, the "Library" menu is activated in Appendix B-9, the "Documents" application's main page. Here all kinds of data are uploaded to a user defined folder structure. Again there are two folders, "Meetings" and "Literature" which contain two Open-Document text-files (*see Appendix B-10*), a pdf-file (*see Appendix B-11*) and a zip-folder with the project aim: the final code (*see*

Appendix B-9). The previously mentioned “Library” menu offers RSS-feeds and functionality to export content to Microsoft programs like Excel or Outlook. The “Modify View” button provides very detailed configuration options for the libraries appearance and operating mode, like filtering or sorting the content.

The next application is the group calendar. Its main page can be found in Appendix B-12. In the page’s centre the actual week is displayed, the whole month is provided in a small version on the left side. SharePoint’s group calendar is a very powerful tool and capable of room-planning, checking members’ calendars, categories for entries, repeating events and the automatic creation of SharePoint workspaces for calendar entries. An exemplary entry is displayed in Appendix B-13.

The “Links” application is very similar to Alfresco’s. Appendix B-14 shows its main page. Three links are added in this scenario. When adding a link the URL, a title and a description are required.

Again the last application is the discussion board. It does also not differ very much from the one Alfresco provides. The main page shows the different topics (*see Appendix B-15*) and an exemplary discussion is displayed in Appendix B-16.

In a short conclusion it can be said that SharePoint offers more functionality than Alfresco does and goes deeper into the different functions. But this larger range of options comes at the expense of the user interface’s clarity and a consistent navigation. A deeper analysis will be done in chapter 6 – Findings and Conclusion.

5.2.2 Application of Testing Scenario 1

Again the project ended and the site is ready to be archived. The following chapter describes the opportunities SharePoint provides for this intent.

The first noticeable fact is that SharePoint does not offer the possibility to simply download a wiki page, blog post or discussion topic in a simple format as html what makes it hard to do a quick copy of the information on a page. The “Shared Documents” and “Documents” libraries provide download-buttons for all files. It is notable that almost all applications provide a button to send the actual page’s link via e-mail but this function only supports sharing content but not storing or exporting it.

But there are ways to store complete sites. A first approach can be to save the site as a template. Templates were already mentioned in chapter 5.2.1 when creating a new site. Just as the first scenario’s template was a “Group Work Site”, a saved site can be a template. All users having the “owner”-rights on the site can perform this action. To do so the

“Site Actions” button from figure 5.13 has to be used to open the drop down menu with the button “Site Settings” while the site that has to be saved is opened. This is important because SharePoint automatically opens the settings page for the current site. Due to SharePoint sites being arranged in a hierarchical order, the server settings can be accessed by the administrator pressing the settings button on his homepage (*see Appendix B-17*). The overview page for the testing site’s settings can be seen in Appendix B-18. The button “Save site as template” opens a dialog window which asks for a file name for the new template, a name that is shown to users and a description of the template. Besides there is an optional checkbox “Include content” which is important here (*see Appendix B-19*). Activating this checkbox ensures that the site’s content is part of the template too. After clicking on “OK” the site can be found as a template under the flag “Custom”.

So this is a first possibility to save a site for later access, but it has some disadvantages: the first one is that only a user with owner rights on the specific site can perform it, making it impossible for non-owners. The second one is that after saving the template all users owning the right to create new sites can use this template. This leads to the problem that there is no access control on the data anymore.

The next approach avoids this problem as it exports the site into one file. Microsoft uses its own archive file format cmp (Content Migration Package). The first problem is that this task can only be performed by a SharePoint administrator, because access to the SharePoint administration interface is needed. Its home page can be found in Appendix B-20. Clicking on the headline Backup and Restore opens all available functions for exporting one or more sites (*see Appendix B-21*). The button “Perform a site collection backup” enables the administrator to back up a site collection, which means all sites on a server. As this is not requested in this scenario the button “Export a site or list” is needed. Appendix B-22 shows the export window which is explained in the following: At first the site-collection has to be chosen. As there is only one server in this case it can be left as default. The second field is the important one: it defines which site shall be exported. The last field empowers the user to define a specific application (wiki, blog, documents etc.) to export. “No selection” determines to capture them all. After specifying the site a file name has to be entered with a path on the server’s hard-disk. Besides it can be chosen to overwrite existing files containing the same name. The point “Export full security” defines if information like author and editors shall be exported too and is checked here. The last option states if all or only the latest versions shall be exported. To do a complete copy of all content all versions are chosen. The button “Start Export” triggers the event and opens an overview page with all information about its status (*see Appendix B-23*).

There is also a feature quite similar to Alfresco's rules: the "Content Organizer". This feature has to be activated via the "Manage site features" button inside of the site settings (*see Appendix B-18*). Once activated the buttons "Content Organizer Settings" and "Content Organizer Rules" appear on the settings page. An exemplary rule for exporting created wiki pages with all its options is stated in Appendix B-24. Again the impression is given that SharePoint and Alfresco both base on the same ideas of functionality but Microsoft's approach goes deeper. The first fact that undermines this impression is the existence of rule-priorities. These determine certain rules to take affect before another one does. The next option defines the content type to export. This example choses the group "Document Content Type" and type "Wiki page". All standard content types like documents, new pages or pictures and also self-defined types can be used here. Next the condition for this rule to take effect has to be determined. Again various possibilities can be defined concerning document type, its name or its content. To ensure all wiki pages to be copied the condition "Name is not empty" is used, as every wiki page must have a name. At last the target has to be chosen. The condition for choosing another site than the source as target is that the Content Organizer is activated there too. In this scenario a new site named "archive" was created and its wiki chosen as target. Besides the target application has to know the exported content type which for example means that a wiki page cannot be exported to the shared documents library. But just like in Alfresco the page is only copied and not archived or applicable in another form.

Of course SharePoint is also able to handle business records and again the functionality has far more range in the commercial tool. Next to the possibility to use a records management site just like in Alfresco, SharePoint provides the possibility to configure integrated records management for every single site differently.

At first the records management functionality has to be activated for the whole server. Appendix B-17 shows the site settings page where under the link "Site collection features" the "In Place Records Management" can be activated (*see Figure 5.16*).

Figure 5.16: Activate Records Management (own source)

Afterwards the button "Record Declaration Settings" appears on settings page (*see Appendix B-17*) where a general configuration of the records management capability can be done (*see Appendix B-25*). The first one defines if records still can be edited, deleted or none of that. So SharePoint provides the possibility to deny all changes on any record

the whole server. But as processing this restrictive is not always reasonable for an enterprise, similar to Alfresco a functionality called “Holds” is provided and will be explained later. The second point in configuring records management states if it is configurable for single applications or only for a complete site and the last one defines who is able use this functionality.

After applying these configurations, the button “Declare Record” is shown in some application’s top menus. So now all uploaded files, documents, posted links and discussion can be declared as record directly within the team room. The other libraries do not have this button so another, not so comfortable way has to be gone. Via the button “Site libraries and lists” on the settings page (*see Appendix B-18*) each library of the site is listed and can be configured. The settings for the wiki library are displayed in Appendix B-26. Under “Record Declaration Settings” the box “Automatically declare items as records when they are added to this list” can be checked (*see Appendix B-27*). This causes that every wiki entry is directly declared as a record. This process can be adapted on any other library.

SharePoint’s “Holds” functionality is again very similar to Alfresco’s. In cases of litigation a record can be put on hold which means it cannot be changed or deleted until it is removed from hold. To access this feature the specific record’s compliance details page (*see Appendix B-28*) has to be opened via a drop-down menu next to the record-file (*see Appendix B-29*). This page provides the hold status for the file and a button to add or remove records from hold.

The retention of records and non-records can be configured for each library via the button “Information management policy settings” which is placed on the specific library’s settings page (*see Appendix B-26*). Here different stages of retention can be defined. Figure 5.17 shows an exemplary retention rule which moves all records into the recycle bin when they become older than three years.

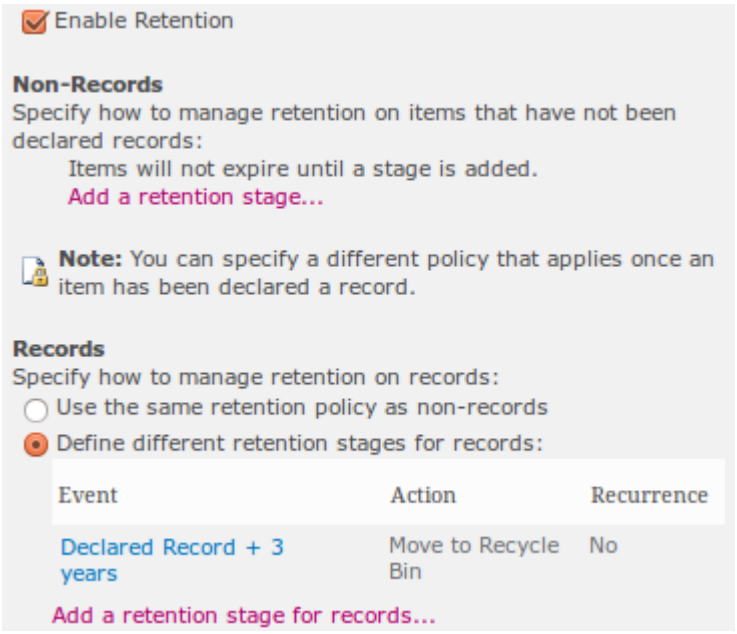


Figure 5.17: Exemplary retention rule (own source)

Even though it is much easier to declare records in SharePoint in general, a direct export of a site into the records management site is also not available. So just like in Alfresco the site backup file has to be uploaded and marked as record manually inside the team room (*see Appendix B-30*) or, if a member of the records management site, using this one (*see Appendix B-31*).

This chapter reveals that both tools do not differ very much in their approaches to export data. SharePoint offers similar features for capturing a whole site into one file as Alfresco does. The records management capabilities are more deeply connected into team rooms but still have some difficulties.

5.2.3 Application of Testing Scenario 2

The scenario in chapter 4.3 defines the task to restore content posted in a site's discussion group, which is closed by now.

Following the first approach in the previous chapter – saving the site as a template – the first condition for this process is that the specific user owns the right to create new sites. If this is not the case the detour over another user with the appropriate right has to be made. As the creation of a site was explained in chapter 5.2.1 the only difference here is that the site backed up is chosen as template. Figure 5.18 displays the new flag "Custom" in the template selection area. All templates created by users appear here.

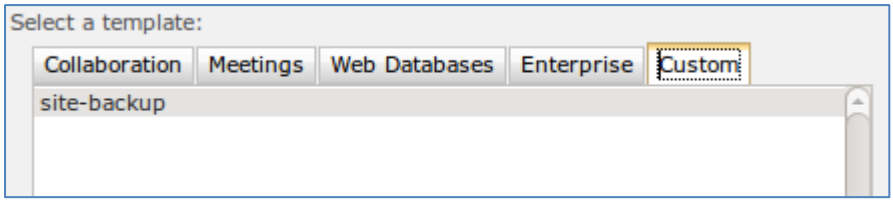


Figure 5.18: Site recovery from template (own source)

After creating the new site it is directly filled with all content that existed at the exporting date. So now the user can access the discussion board as he is used to and retrieve the information he needs.

Backed up sites (cmp-files) which are declared as business records also have to be downloaded and cannot be restored into SharePoint from the Records management site. Restoring a site out of a cmp-file is a more difficult task than using a template. This process requires administrator rights on the platform and access to the SharePoint server. This is necessary because the import has to be triggered via the Microsoft SharePoint 2010 Management Shell – a command line interface. These interfaces generally provide powerful configuration and management tools for all kinds of tasks to perform. Their advantage is that via one single command complex processes can be triggered that would require very large graphical user interfaces. It can be started just like it is known from Windows via the Start menu → All programs → SharePoint 2010 → SharePoint 2010 Management Shell. The different commands can be found in Microsoft's SharePoint online documentation (Microsoft 2011). Before importing a site some arrangements have to be done. At first a new empty site has to be created. This is the target site and it has to use the same template as the exported site does otherwise the Management Shell will throw an error. As parameters the shell command needs the target site's URL ("-identity") and the path to the backed up cmp-file ("-path"). The exact command to import the file "Site3.cmp" created in the previous chapter reads as follows:

"Import-SPWeb –identity http://nils-pc/backup -path C:\\backup\SharePoint\Site3.cmp"

The display after triggering this command can be found in Appendix B-32. If the process completes without errors the path to the log-file is displayed. Beneath the shell the end of the created log-file is shown, which states time and date, duration and imported objects (*see Appendix B-32*). Besides, the number of warnings and errors is shown. Given that the import succeeded there are no errors. The warnings refer to old test users that cannot be resolved anymore. Now users with access to the specified target site can retrieve all content from the old site, again with exception of group calendar entries.

Summing up the retrieval of backed up SharePoint sites is a quick task for an administrator who knows where to import which backup file, but turns out as impossible for normal users, except the template-approach is chosen (which is not recommended because of the mentioned reasons).

6 Findings and Conclusion

The final chapter will summarize the findings of chapter 5 and give a detailed discussion to which degree the research questions have been answered. At first the outcome of Alfresco's and SharePoint's testing processes will be displayed in a figure and then analysed and compared. Afterwards a final discussion on the results and future research to do is done.

6.1 Solutions

This chapter provides the outcome of chapter 5 as a visualisation.

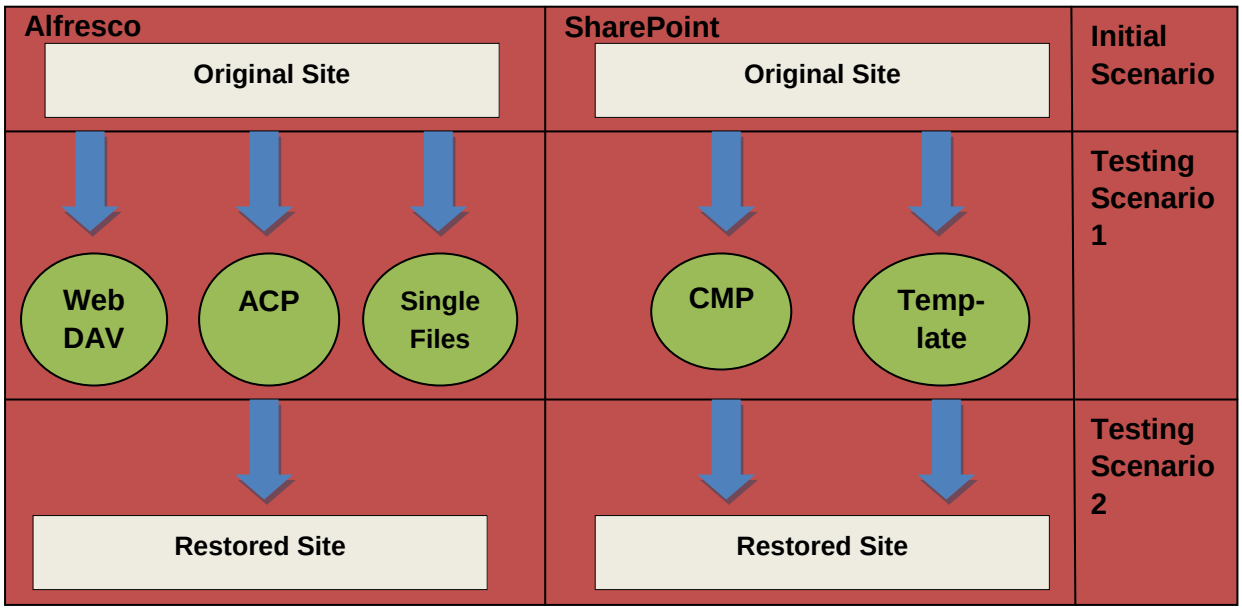


Figure 6.1: Visualisation of solutions (own source)

The figure is split into three rows which stand for the different scenarios. As explained, the first scenario creates the basis for next steps. The next row displays the different approaches to export content from the systems. Indeed, Alfresco supports more ways to get content out of the system than SharePoint does, but as the figure shows only one of them is suitable for retaining a whole site in the last scenario. However, SharePoint's two ways of archiving a site both have a corresponding retrieval function for the next scenario.

6.2 Analysis

The analysis of the scenarios' outcome is split into three chapters. Each of them focuses on one scenario and compares the different solutions primarily considering functionality and usability. Besides, attention is spent on the tool's transparency and openness.

6.2.1 Initial Scenario

As this scenario was used to create the same basis for the tools it did not intend to answer a research question. Instead the population of the tools with content was used to get an overview of their range and the way they are used. The following table displays the main aspects of chapter 5.1.1 and 5.2.1 in short form and is explained afterwards.

	Alfresco	**SharePoint**
Functionality	• Contains all basic functionality for team collaboration by default • Basic requirements fulfilled by individual applications, but not penetrative • Few customization possibilities	• Extensive amount of functionality • Highly customizable • Partial desktop integration
Usability	• Quick learning process • Easy and completely consistent navigation in all applications • Simple invitation of new users	• Partial inconsistencies in the navigation • Not always intuitive, but practicable • Complex user rights configuration

Table 6.1: Findings – Initial Scenario (own source)

While executing the first scenario the difference between both programs gets clear very quickly: Alfresco puts its focus on simple user interfaces and easy usable applications, while SharePoint provides a widespread spectrum of applications for all kinds of use. The first example supporting this assumption is reflected in the creation of a site: Alfresco demands five inputs: name, URL, description, type and visibility (*see Figure 5.2*). By default only the type "Collaboration-Site" is available. In contrast SharePoint provides by default over 20 different site-templates (Appendix B-3 displays only six, the other ones can be accessed through the other flags). The next example is the site itself. An Alfresco

site has nine default applications and cannot be extended without installing new packages while SharePoint sites contain many libraries which can easily be added (*see Appendix B-1*).

Creating new users is another good point. The Alfresco administrator can simply invite users via e-mail while the SharePoint environment uses the Windows domain's users. This makes SharePoint more inflexible but also gives more control over the users to the administrator. The user rights definition is also far more configurable. All Alfresco users – with the exception of the administrator – generally have the same rights, for example to create a site. The SharePoint rights management provides the possibility to create user groups with different available functionality what is not possible to this extent in Alfresco.

But as already mentioned in chapter 5.2.1 SharePoint's high level of configuration options and functionality have also negative aspects. The first one is that it is not possible to display all buttons and features at once which means that the user has to switch between different menus inside of one application. As the different applications contain different functionality these menus have different looks. Thereby the recognition value is very little and it is difficult for users to remember where the functions are placed. Coming along are the inconsistencies in the navigation. While generally all applications are used via the top menu, the blog functionality is provided in a right sidebar. This also complicates the learning process. However in Alfresco all applications provide their functionality exactly in the same top menu bar and the navigation through the created content is always placed in the left sidebar.

All in all Alfresco presents itself as a light-weight and user friendly system while SharePoint convinces with many features and customizations capabilities.

6.2.2 Testing Scenario 1

This chapter summarizes the solutions from chapter 5.1.2 and 5.2.2 and answers the first research question:

1. How is the archiving of web-content integrated into ECM-tools (on a functional level)?

The main findings are presented in a table at first again. This time the table is extended by the row "Transparency" as this scenario lends itself to investigate in this point.

	Alfresco	SharePoint
Functionality	• Every piece of content downloadable as one file • Site export into a single package • WebDAV support • Records management capabilities (with export functionality)	• Saving sites as templates • Site export into a single package • Comprehensive records management functionality
Usability	• Export functionality applicable for each user • Use of second web interface (Alfresco Explorer) confusing in the beginning • Records management not integrated into team rooms	• Export functionality applicable only by few users • Package exported directly to the SharePoint server (no access via web interface) • Records management functionality integrated into team rooms
Transparency	• Good insight into the tool's backend • Use of open standards and formats	• Closed up system • Some components only support Microsoft's file formats

Table 6.2: Findings – Testing scenario 1 (own source)

At first it has to be said that both tools solve the given task in very similar ways. So, answering the first research question, on one hand the systems provide functions to export complete sites into one package and on the other hand a business records capability – which is in both tools implemented as a site – is available to store this package. The packages are both of an own archive-file type which can only be reasonably used by the appropriate system. Besides Alfresco provides a special export button at all levels of the records management site which is another simple way to capture important content – if it was declared as a record before.

But according to file types this chapter is a good example to display some advantages that Open Source solutions can have over proprietary software. Alfresco presents itself as completely transparent regarding the point how it stores content that is created in team rooms. Each user can access a repository that provides all content in a hierarchical folder structure. All content that is created directly in Alfresco – that are wiki-pages, blog posts, links and discussions – are saved as html-files and can be downloaded. In contrast SharePoint does not provide any possibilities just to identify the file type that is used for this content. The "Shared Documents" functionality can only be used to its full extent when Microsoft Office is installed on the user's computer. Anyway, the use of html may be a simple way to store content, but outside the system it is very impractical.

The next advantage is the use of the open standard WebDAV. Via this interface a whole site or only single applications can be copied directly from the Alfresco server by every user according to his rights. In SharePoint a site-member without manager-rights has no possibility to archive a site, what results in strong dependencies between users when content has to be archived or retrieved.

While the exporting functionality does not differ very much, SharePoint's business records capabilities are much more integrated into the team rooms. That means it is possible to declare single files as records inside a team room. But again the tool has to struggle with inconsistencies: wiki pages and blog posts cannot be declared as records this way; here rules have to be set.

6.2.3 Testing Scenario 2

The last chapter of analysis answers research question 2 according to the findings of chapter 5.1.3 and 5.2.3:

2. Which ways for retrieving archived web-content are provided?

Again a table summarizing the findings is provided.

	Alfresco	**SharePoint**
Functionality	• Importing acp-files	• Importing cmp-files • Creating sites from templates
Usability	• Import applicable by each user via web interface • Content is not available in the original application	• Import only per command line interface by administrator • Original site completely recreated (except of calendar

	→ access via Alfresco Repository	entries)

Table 6.3: Findings – Testing scenario 2 (own source)

The last scenario is again an example for Alfresco's user friendliness. Each user is able to import a site via the web interface – given that he has access to the package file. The only problem is that importing a site does not recreate the content in its appropriate applications, but creates a new subfolder similar to the single application-folders. So the content is only applicable through the Alfresco Repository.

SharePoint has a more elegant approach: importing a site creates a site equal to the original one with exception of calendar entries. But this solution requires the use of a Management Shell – a command line interface – which is a powerful administration tool for administrators, but again the task is not applicable for usual users.

The use of site-templates is indeed very simple but this functionality's aim is not the archiving of sites, but providing pre-configured team rooms as a starting point. As a template that is saved from a site contains all of its content, all users with site-creation rights have access to it, which is no desirable state for an enterprise.

The last scenario also supports the assumption made in chapter 6.2.1: Alfresco supports all the basic functionality for administering content in online team rooms in a user friendly way and is suitable for the use in small projects, while SharePoint is an all-round system supporting all kinds of online team work and collaboration.

6.3 Conclusion and Future Work

The thesis' last chapter provides concluding thoughts on this research's outcome. To do so the findings from chapter 6.2 will be applied on enterprises' problems and requirements, raised in the first chapter. In addition it is discussed what future research on the archiving of web-content has to be done.

At first it has to be mentioned that this thesis only tested the standard systems. Both can be expanded by additional functionality, programmed by other companies or individuals owning a so called source developer kit (SDK) which is in both cases downloadable for free under an Open Source license. This leads to many custom-programmed applications that bring additional functionality. But this independency from the original programmers also creates problems like longer waiting times for some functionality to be available in newer versions of the tools. A good example is the Alfresco extension "Information Lifecycle Management" by Westernacher Products and Services AG (Alfresco Ltd.

2010). This expansion brings several new records management and archiving features to Alfresco that would support this thesis aim but is only available until version 3.3 which is already two versions previous to the current.

Now Kampffmeyer's assumption from chapter 2.3, where he illustrates the web-content-archiving as an "open topic" (Kampffmeyer 2011, p.6) is picked up:

On one hand he is definitely right. None of the tools presented a perfect solution for every user. On the other hand, commercial software vendors as well as the Open Source community seem to get aware of the need to archive content arising from computer supported cooperative work – especially regarding the fact that many solutions offered by the tools follow similar approaches.

While the Open Source software can benefit from its application of free standards which only works out if the end-user is aware of this and able to handle it, Microsoft integrates business records management more and more into online team rooms. Although this is a step into the right direction because this content also has to meet governmental and regulatory needs, the approach is still not technically mature. Saving wikis and blogs as records requires complex configurations where in contrast the Open Source ECMS does not provide integrated records management, but a simple download function. So a detour has to be made over downloading a file before declaring it a record but this is still more user friendly than the commercial alternative. To meet regulatory and legal needs, for example to solve the problem of making data available in cases of litigation, mentioned by Williams (Williams 2011, p.17) both tools again provide similar approaches: the "freezing" of data to prevent subsequent changes on files.

But this discussion – if making a detour is easier or more user friendly than the complex configurations – supports Kampffmeyer's assumption that the archiving of web-content is not solved by now. The storing of archived team rooms is still depending very strong on the user's, respectively the administrator's motivation in archiving and records management because both tools do not provide full integration of those two topics.

One step to take in future research would be the analysis of further tools. Taking a look at figure 2.3 and 2.4 reveals the tested tools as "follower" (Microsoft) and "challenger" (Alfresco) on the ECM market with low to medium technology assessment. Both figures define IBM and Oracle as enterprises with a very high technology assessment what makes them ideal candidates for further applications of the scenarios.

But already at this point of research new questions closely related to business problems show up: which content arising in an Enterprise 2.0 is worth being a record? Which content must or must not be one? Other than in traditional records management where for

example every single contract signed by an enterprise has to be kept, a wiki in most cases probably does not have to be declared a record in complete but only a few pages of it. Gantz et al. described these questions already in 2007 as hard to answer and spoke of a maximum value of 20% of digital content that has to fall under compliance rules (Gantz et al. 2007, p.13/14).

That shows the importance of new records management policies, otherwise the records management systems will become the new “data-silos” that Williams and Hardy spoke of (Williams & Hardy 2011, p.57).

References

AIIM 2012, What is Enterprise Content Management?, viewed 20 June 2012, <http://www.aiim.org/What-is-ECM-Enterprise-Content-Management>

Alfresco Ltd. 2010, Alfresco Solutions Showcase: Information Lifecycle Management, viewed 16. September 2012, <http://blogs.alfresco.com/wp/solutions/information-lifecycle-management>

Alfresco Ltd. 2011, Wiki: Records Management, viewed 14 September 2012, <http://wiki.alfresco.com/wiki/Records_Management>

Courage, C. & Baxter, K.: Understanding your Users: a practical guide to user requirements, San Francisco: Morgan Kaufmann, 2005.

Eggert, S. & Gronau, N.: Erhöhung der Wandlungsfähigkeit von ECM-Lösungen unter Verwendung kartographischer Gestaltungsmittel, in Bichler, M.; Hess, T.; Kcrmar, H.; Lechner, U.; Matthes, F.; Picot, A.; Speitkamp, B.; Wolf, P. (Hrsg.): Multikonferenz Wirtschaftsinformatik 2008, p.433-446, Berlin: GITO, 2008.

Eggert, S.; Gronau, N. (Hrsg.): Enterprise Content Management, Berlin: GITO, 2007.

Fowell, S.: Bridging the Gap between Information Resource Design and Enterprise Content Management, in Digital Libraries: People, Knowledge, and Technology, 5th International Conference on Asian Digital Libraries, Singapore, p.507-515, 2002.

Gantz, J. et al.: The Expanding Digital Universe, IDC/EMC Whitepaper, 2007.

Gantz, J. & Reinsel, D.: The Digital Universe Decade – Are You Ready?, IDC/EMC Whitepaper, 2010.

Gartner Inc.: Magic Quadrant for Enterprise Content Management, Mark R. Gilbert, Karen M. Shegda, Kenneth Chin, Gavin Tay, October 13 2011.

Gartner Inc. (1) 2012, IT Glossary: Enterprise Content Management, viewed 20 June 2012, <http://www.gartner.com/it-glossary/enterprise-content-management-ecm>

Gartner Inc. (2) 2012, IT Glossary: Enterprise Information Management, viewed 20 June 2012, <http://www.gartner.com/it-glossary/enterprise-information-management-eim>

Hardy, C.A. & Williams, S.P.: Managing Information Risks and Protecting Information Assets in a Web 2.0 era, in 23th Bled eConference eFuture: Creating Solutions for the Individual, Organisations and Society, Bled Slovenia, p.234-247, 2010.

International Organization for Standardization: ISO 15489-1: 2001 Information and documentation – Records management. Part 1: General, section 3.16, Geneva: ISO, 2001.

JITC 2007, Records Management Application - DoD 5015.02-STD RMA Design Criteria Standard, viewed 14 September 2012, <http://jitc.fhu.disa.mil/cgi/rma/standards.aspx>

Kampffmeyer, U.: ECM - Enterprise Content Management, Hamburg: Project-Consult, 2006.

Kampffmeyer, U.: Elektronische Archivierung – Ein ständig aktuelles Thema, online publication, Hamburg: Project-Consult, 2011.

Kelle, U: Die Integration qualitativer und quantitativer Methoden in der empirischen Sozialforschung: Theoretische Grundlagen und methodologische Konzepte, Wiesbaden: VS Verlag für Sozialwissenschaften, 2008.

Microsoft 2011, SharePoint 2010 Products Documentation, viewed 11 September 2012, <http://technet.microsoft.com/en-us/library/ee428287>

Olson, J.: Database Archiving: How to Keep Lots of Data for a Very Long Time, Burlington: Elsevier, 2009.

Österle, H. et al.: Gestaltungsorientierte Wirtschaftsinformatik: Ein Plädoyer für Rigor und Relevanz, Nürnberg: Infowerk, 2010.

Ovum: Technology Components of Content Management, Butler Group, 2010.

Ovum: SharePoint 2010 for Enterprise Content Management, 2011.

Ovum: Enterprise Content Management 2011/2012 – driving business value from Content Management, 2012.

Project Consult 2011, Enterprise Information Management, viewed 20 September 2012, <http://www.project-consult.de/ecm/information-management/enterprise-information-management>

Rapoport, R.N.: Three Dilemmas in Action Research: With Special Reference to the Tavistok Experience, Human Relation, vol. 23, no. 6, p. 499-513, 1970.

Wilde, T. & Hess, T.: Forschungsmethoden der Wirtschaftsinformatik, in Wirtschaftsinformatik 49, p.280-287, 2007.

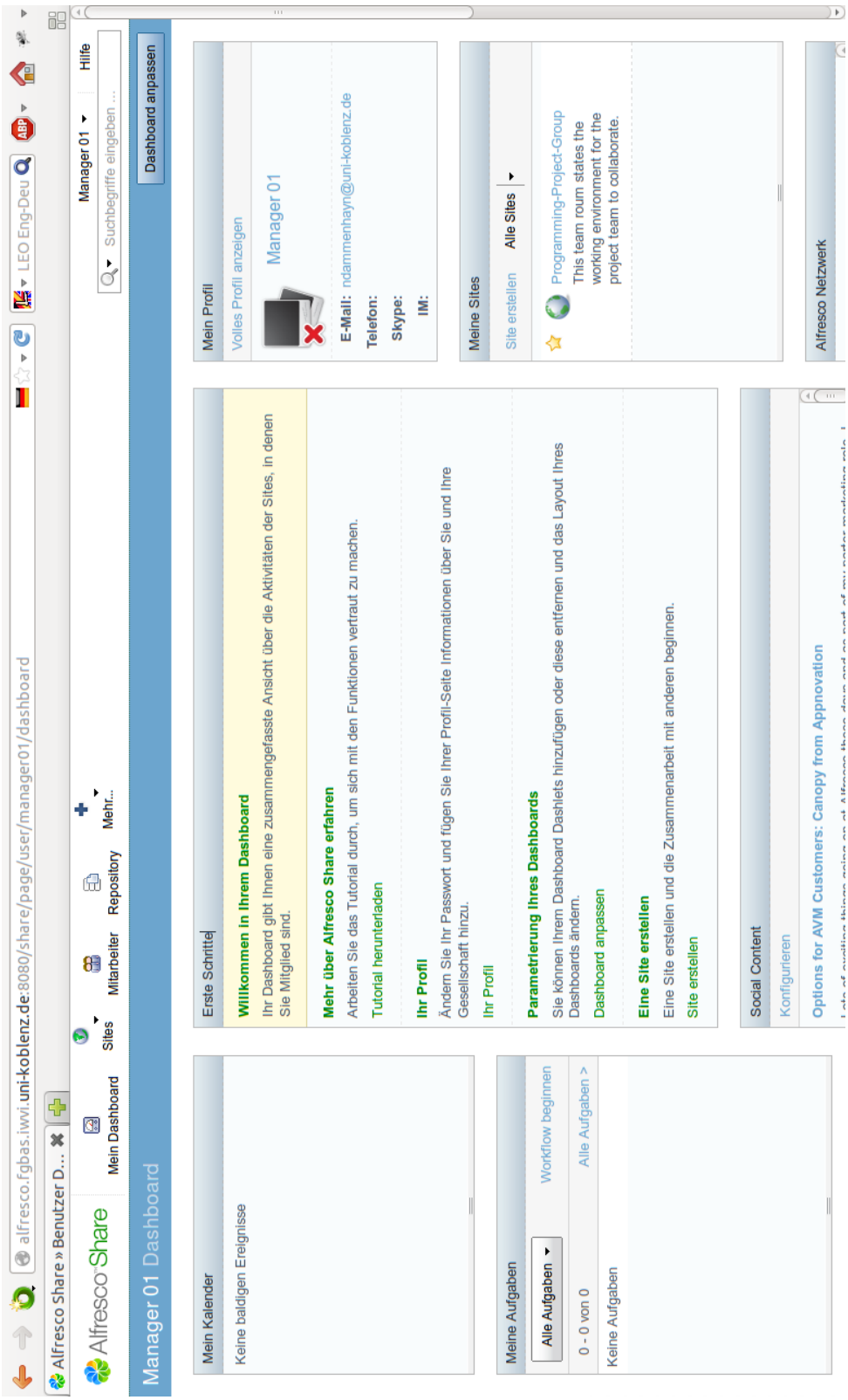
Williams, S.P.: Das 8C-Modell für kollaborative Technologien, in Schubert, Petra; Koch, Michael (Hrsg.): Wettbewerbsfaktor Business Software, p.11-21, München: Hanser, 2011.

Williams, S.P. & Hardy, C.A.: Information Management Issues and Challenges in an Enterprise 2.0 Era: Imperatives for Action, in 24th Bled eConference eFuture: Creating Solutions for the Individual, Organisations and Society, Bled Slovenia, p.56-67, 2011.

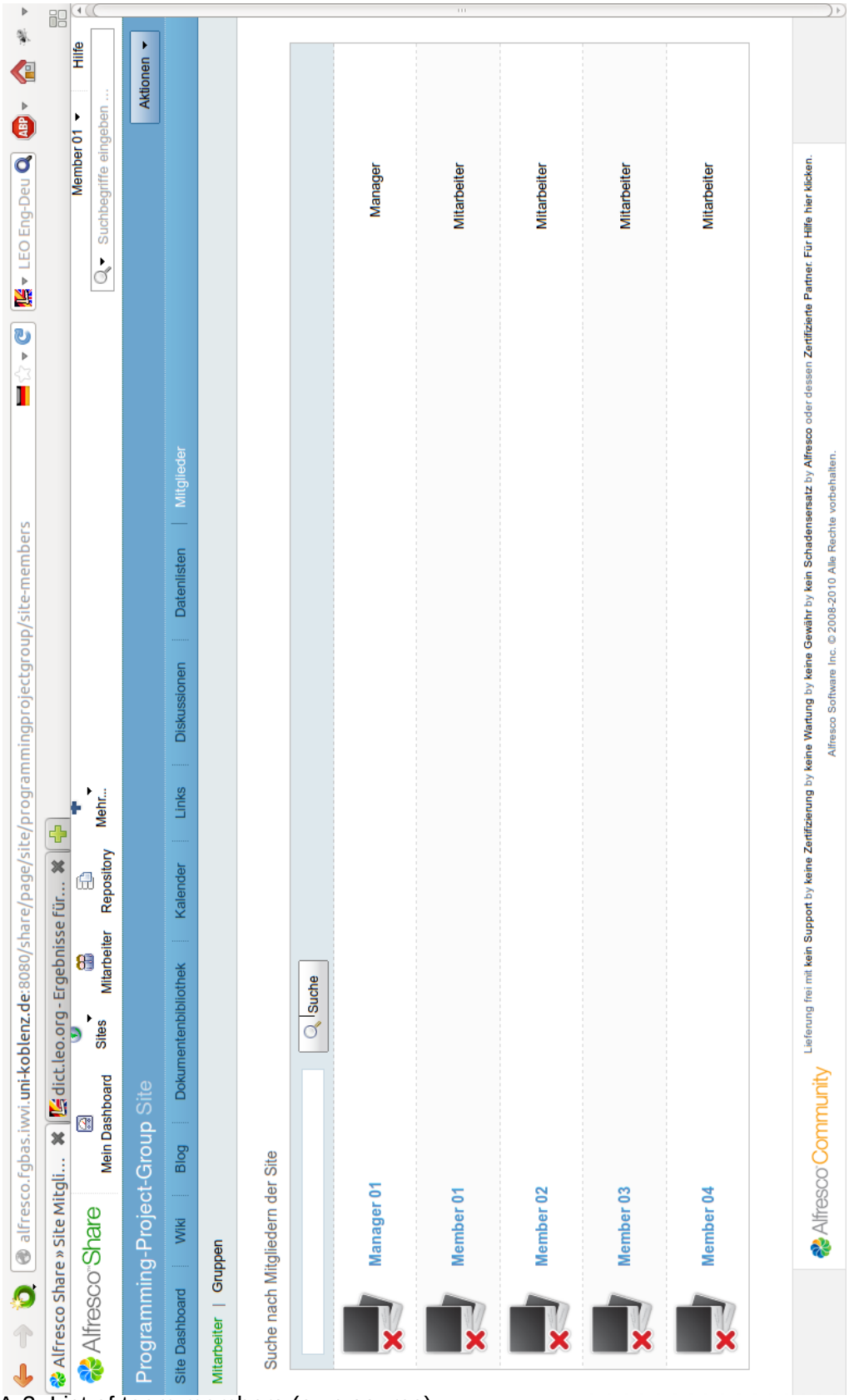
Witt, H.: Forschungsstrategien bei quantitativer und qualitativer Sozialforschung, in Forum Social Research, vol. 2, no. 1, art. 8, February 2001.

Zelewski, S.: Theoretische Fundierung der Wirtschaftsinformatik: Fragmentarische Thesen zu Anspruch und Wirklichkeit aus der Perspektive eines Grenzgängers, in Jung, R. & Myrach, T. (Hrsg.): Quo vadis Wirtschaftsinformatik? Festschrift für Prof. Gerhard F. Knolmayer zum 60. Geburtstag, p.163 – 190, Wiesbaden: Gabler, 2008.

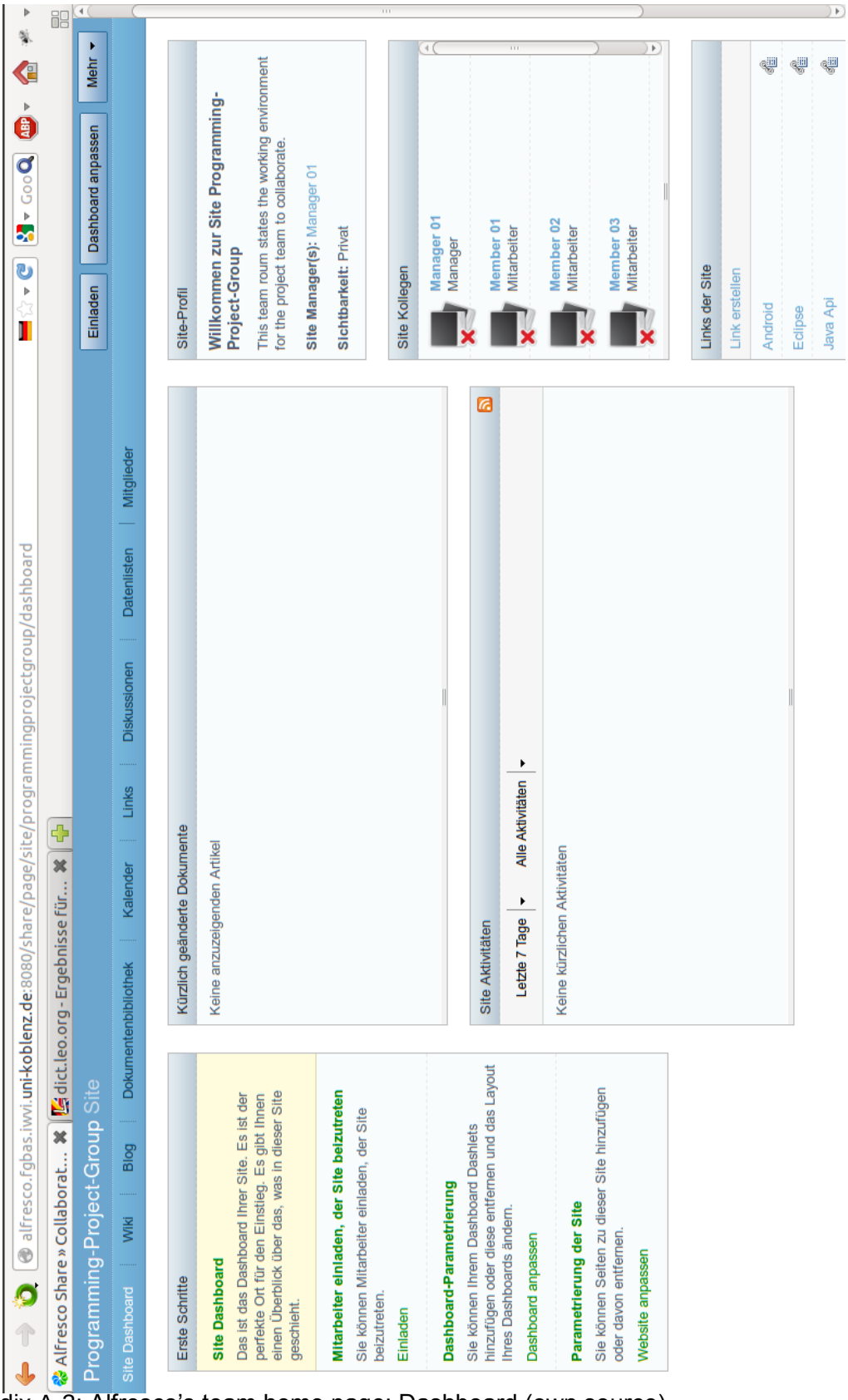
Appendix A - Alfresco



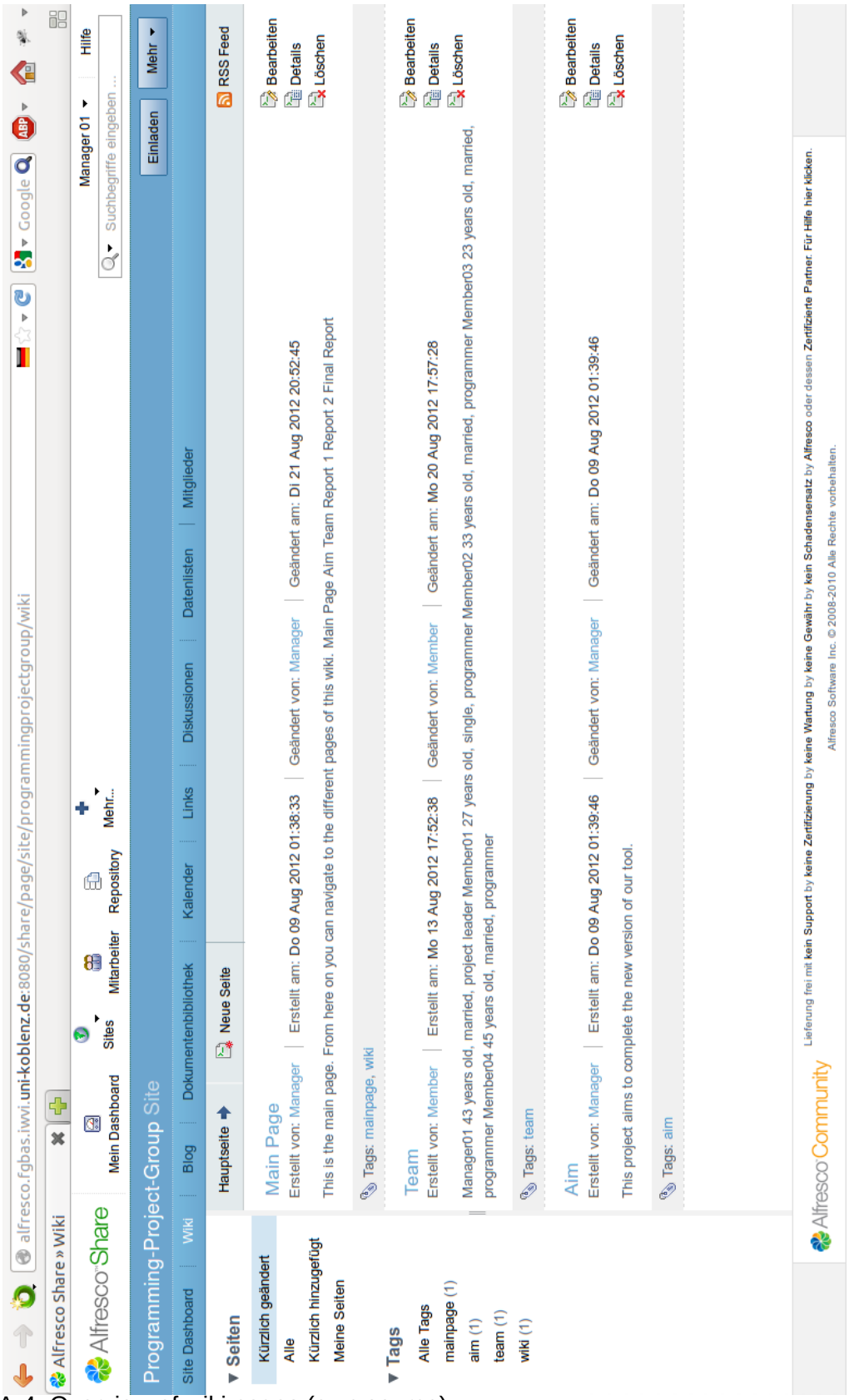
Appendix A-1: Alfresco's home page (own source)



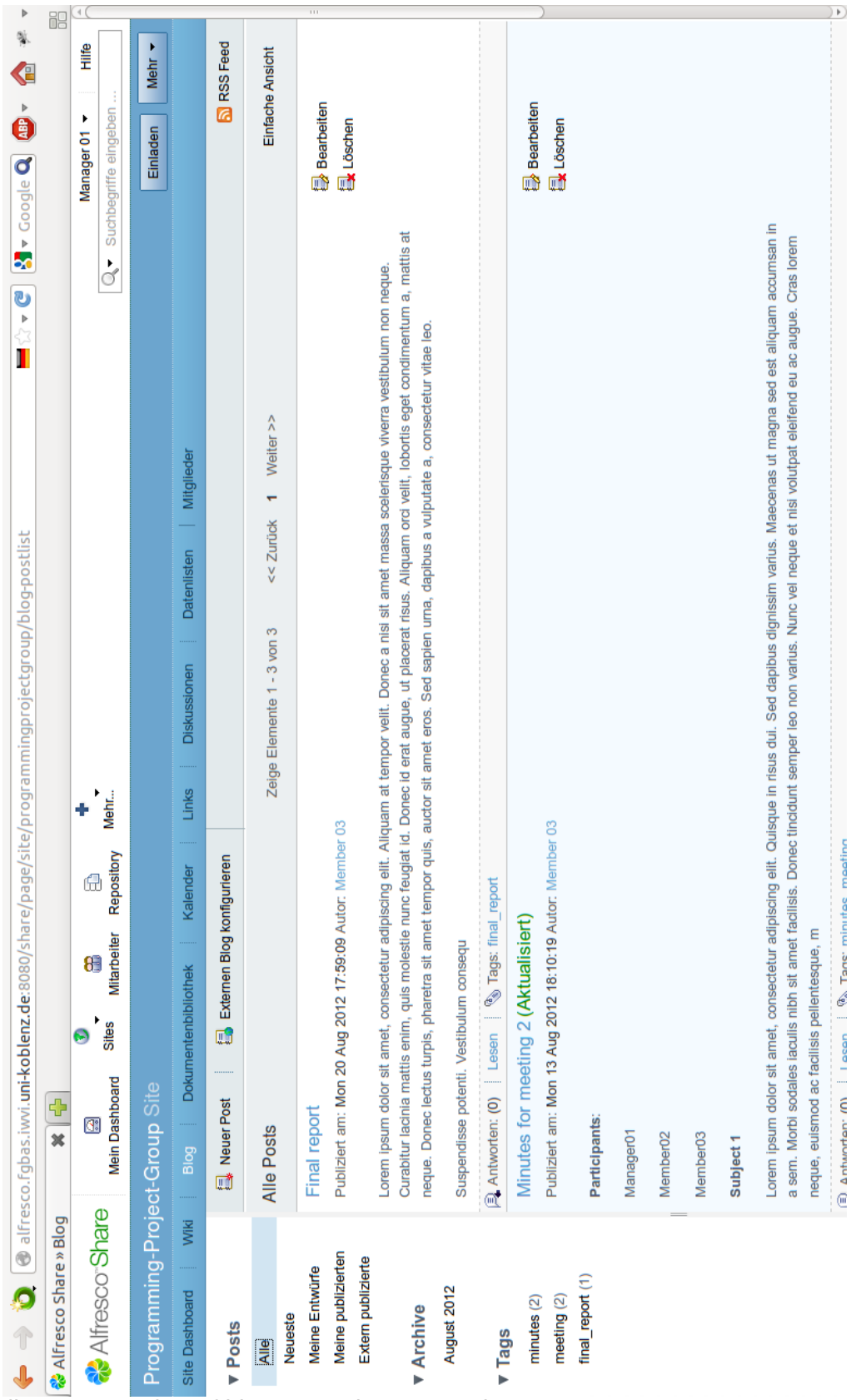
Appendix A-2: List of team members (own source)



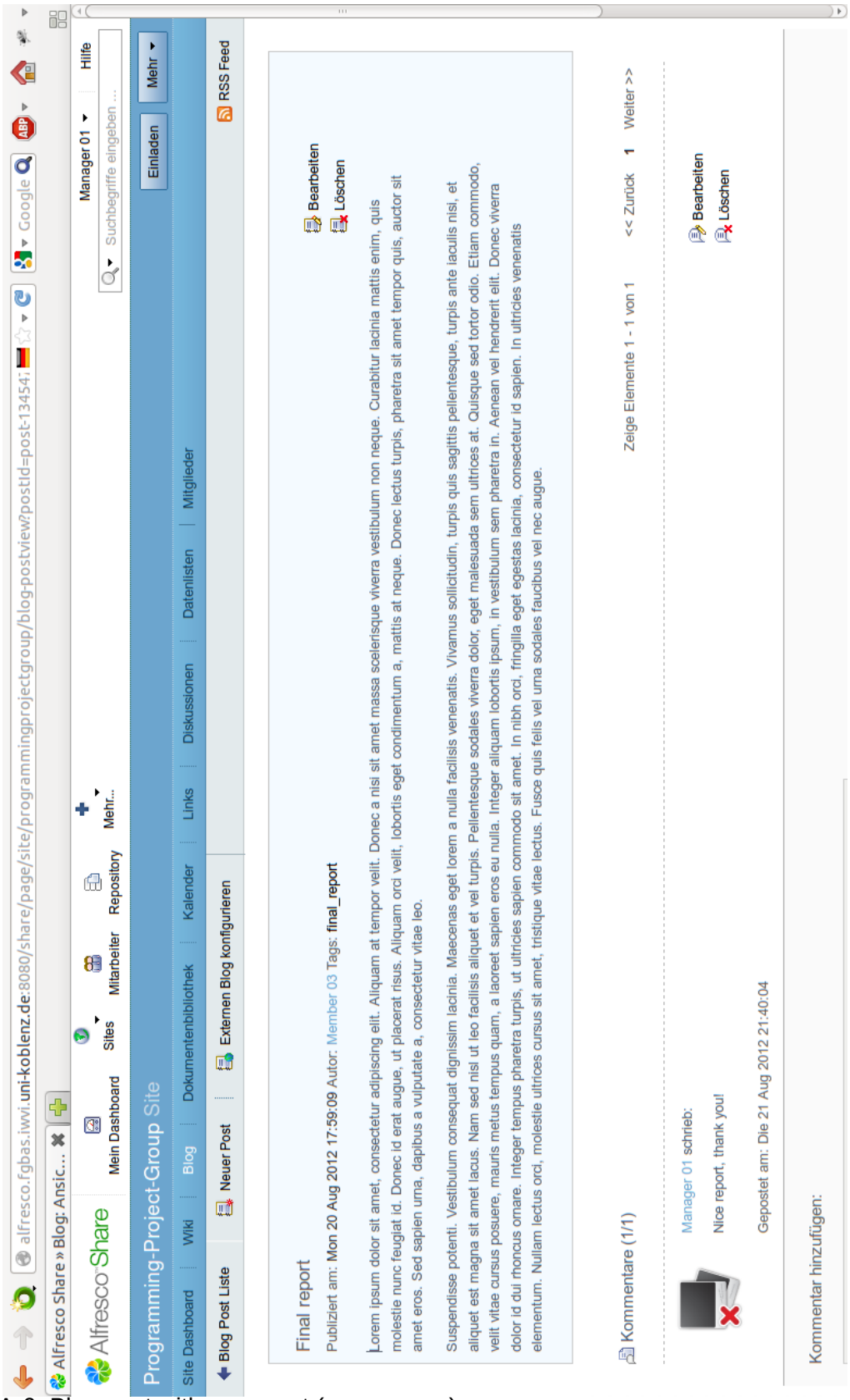
Appendix A-3: Alfresco's team home page: Dashboard (own source)



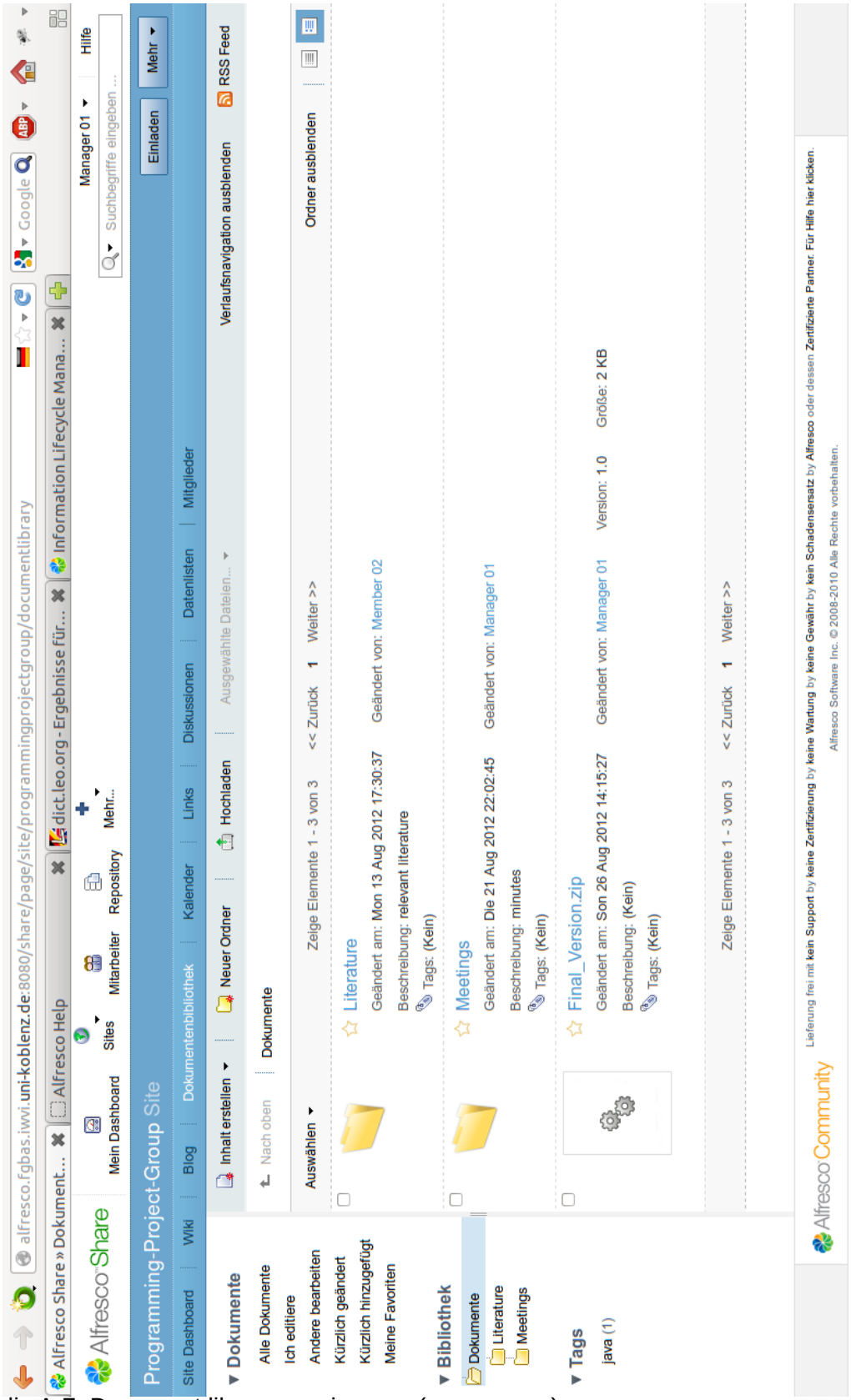
Appendix A-4: Overview of wiki pages (own source)



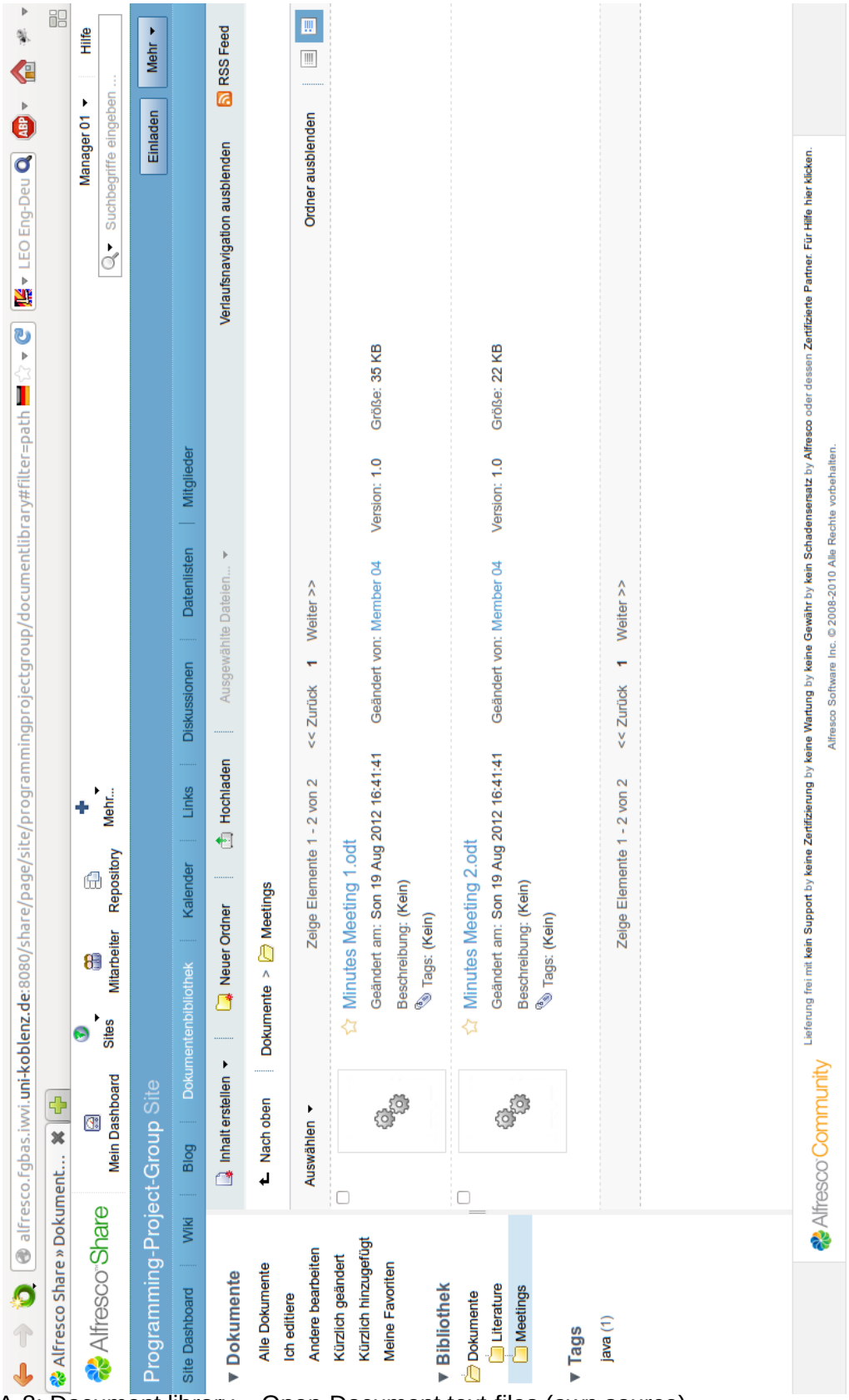
Appendix A-5: Overview of blog pages (own source)



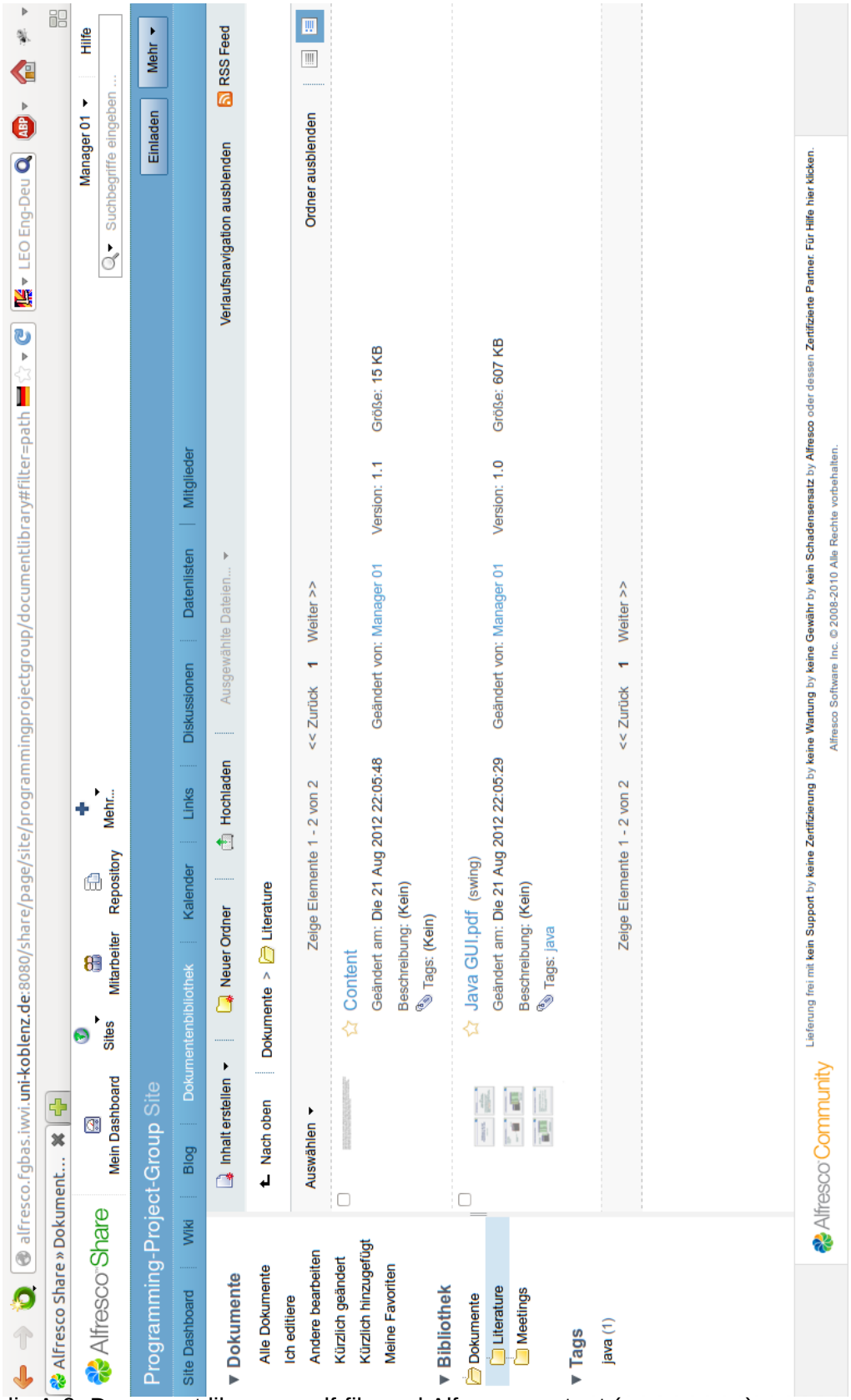
Appendix A-6: Blog post with comment (own source)



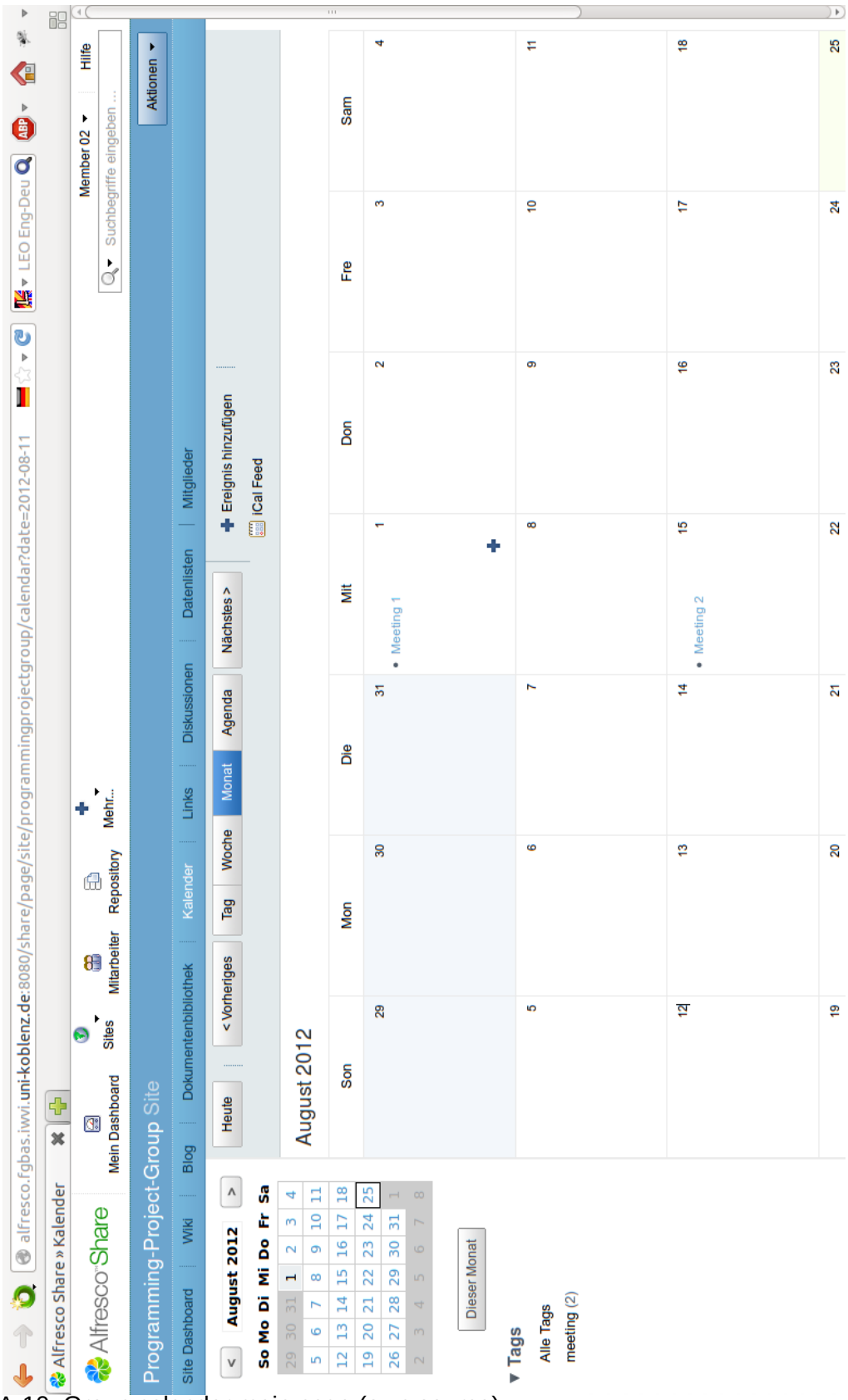
Appendix A-7: Document library - main page (own source)



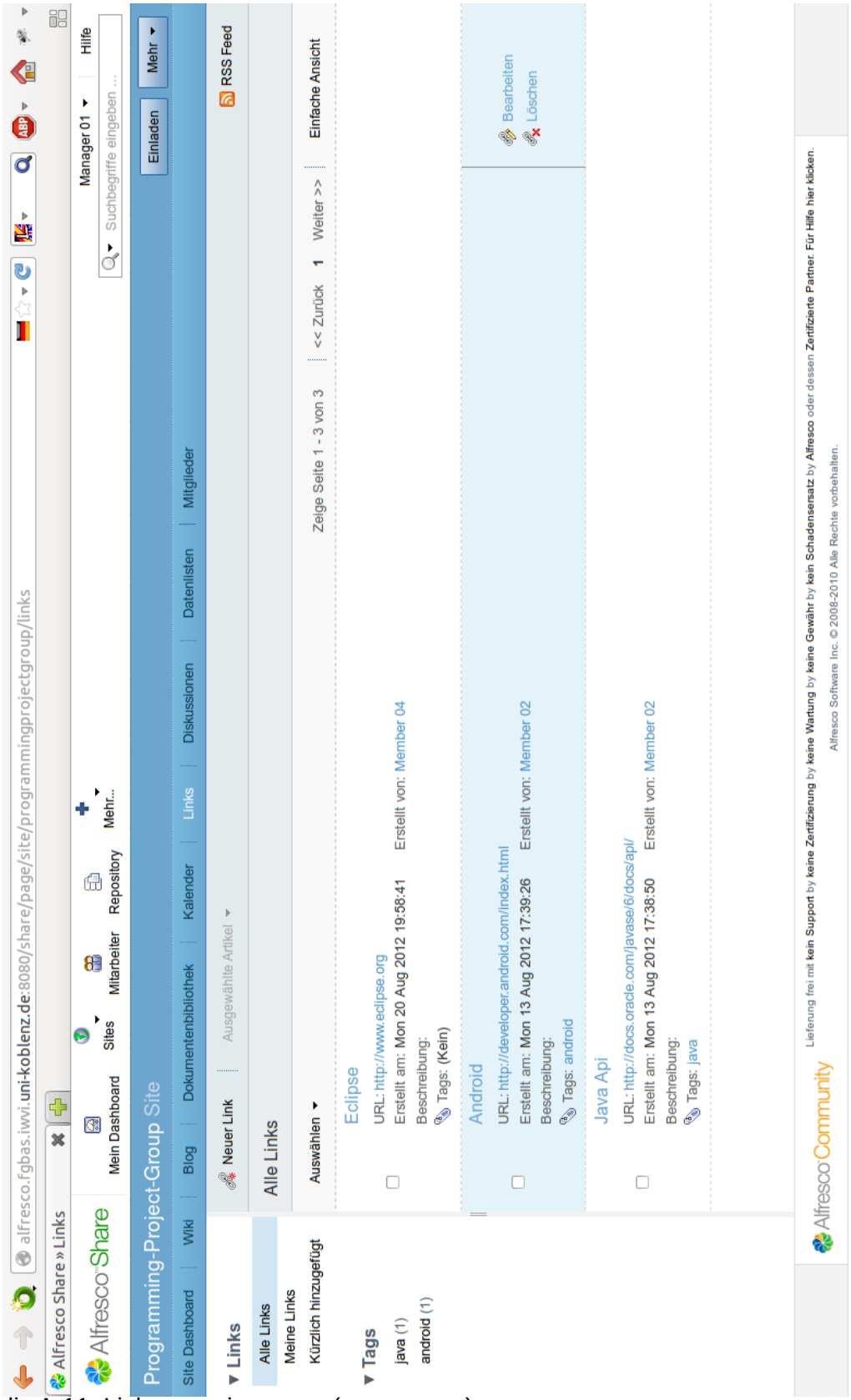
Appendix A-8: Document library – Open-Document text-files (own source)



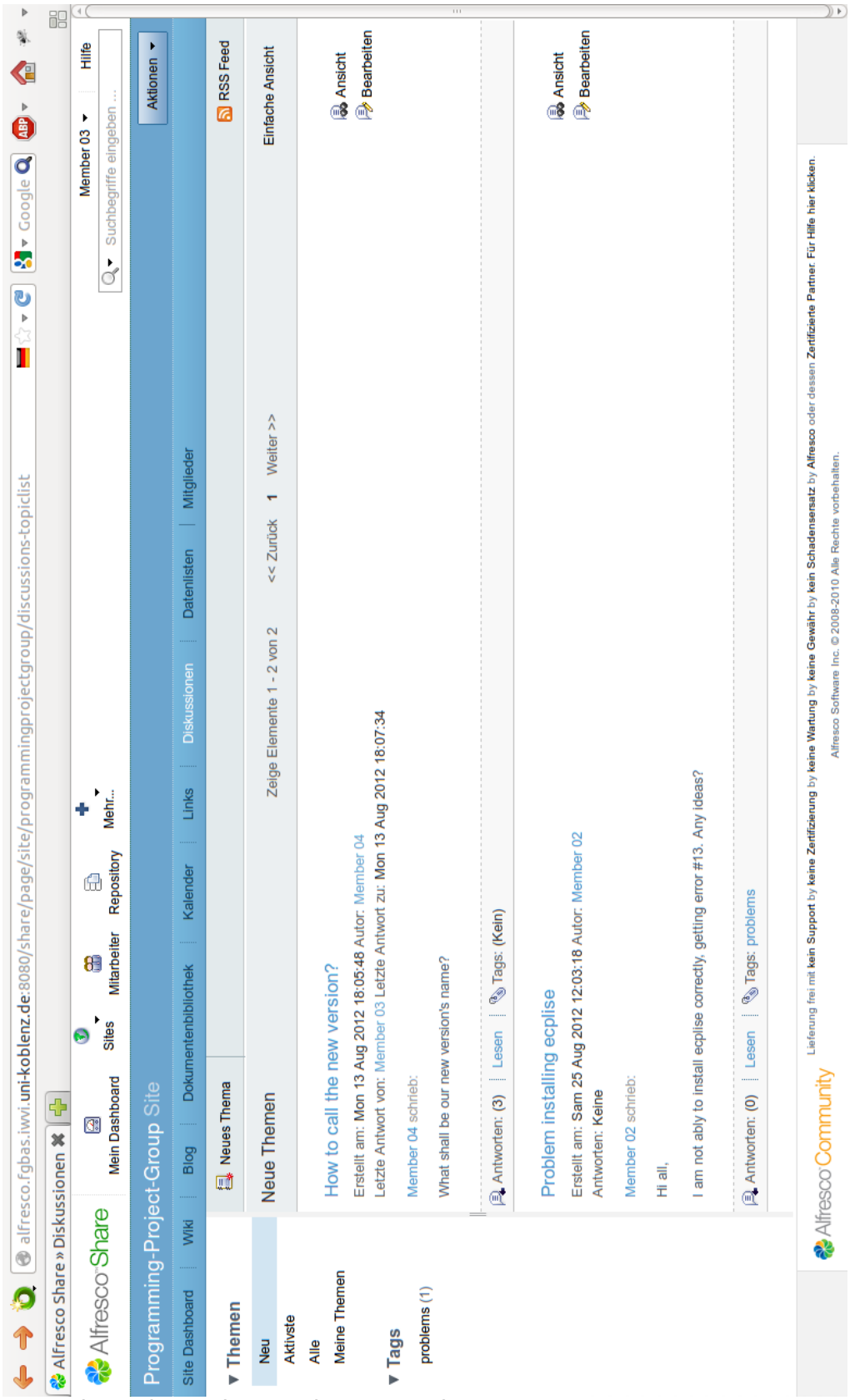
Appendix A-9: Document library – pdf-file and Alfresco content (own source)



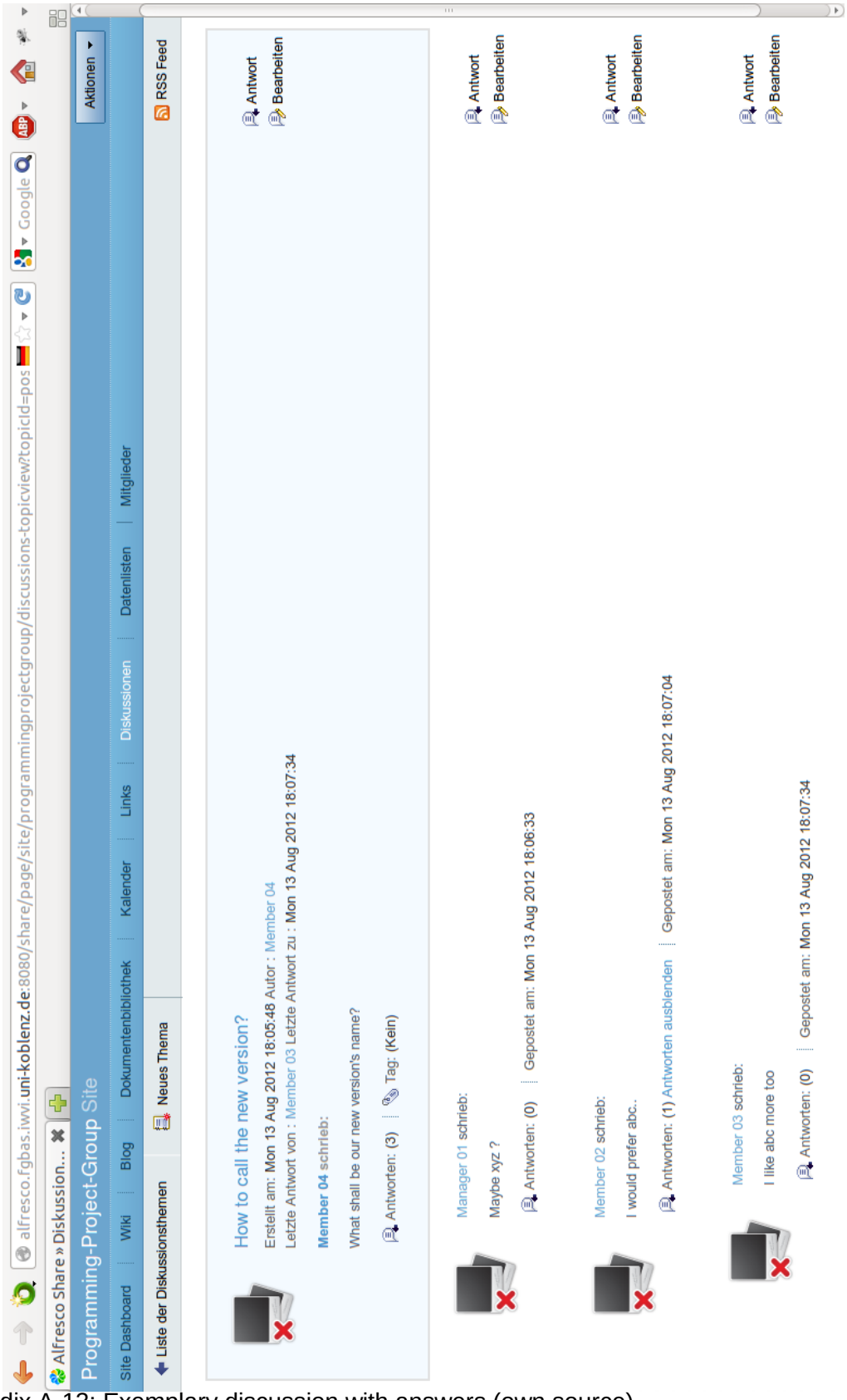
Appendix A-10: Group calendar main page (own source)



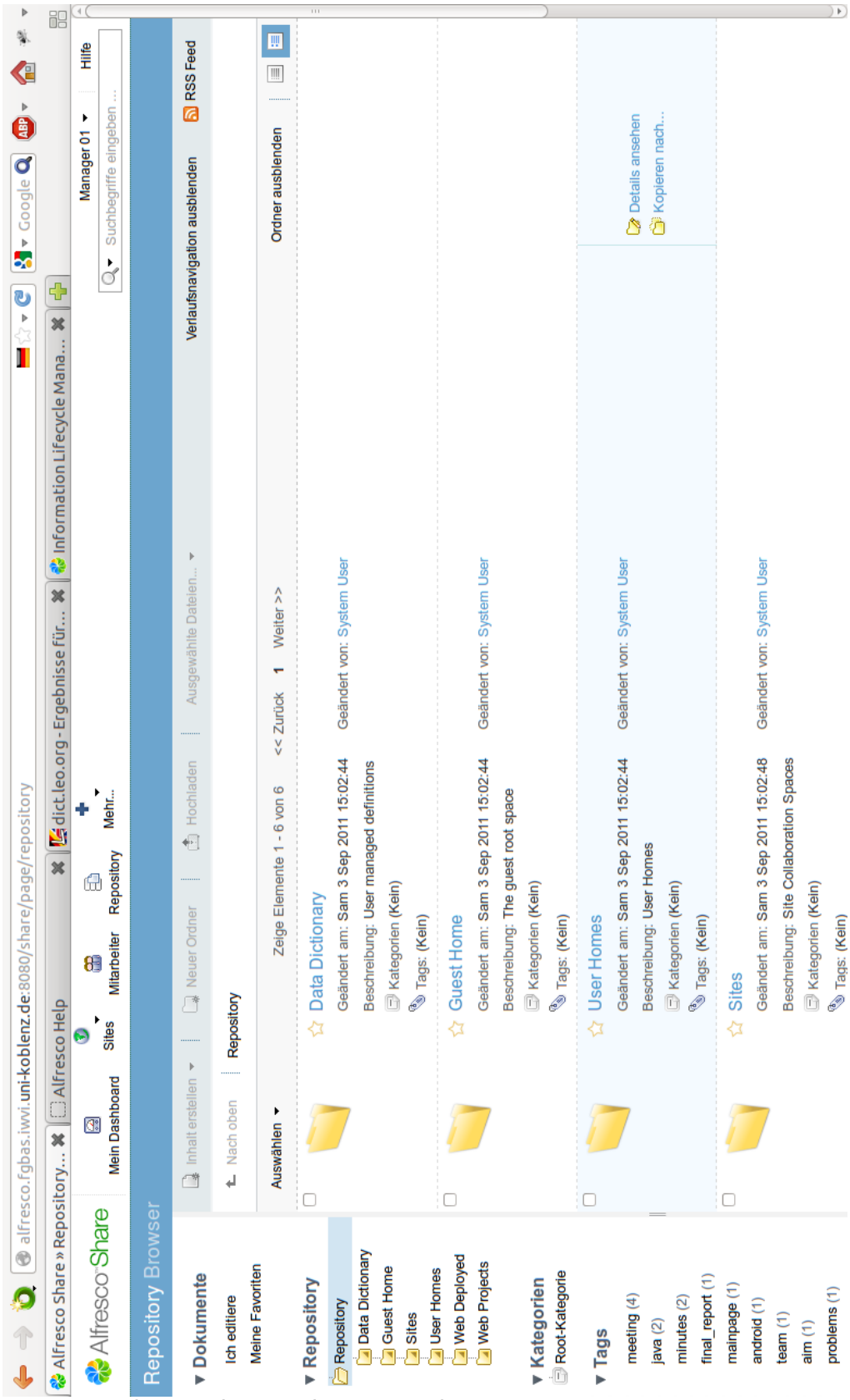
Appendix A-11: Links overview page (own source)



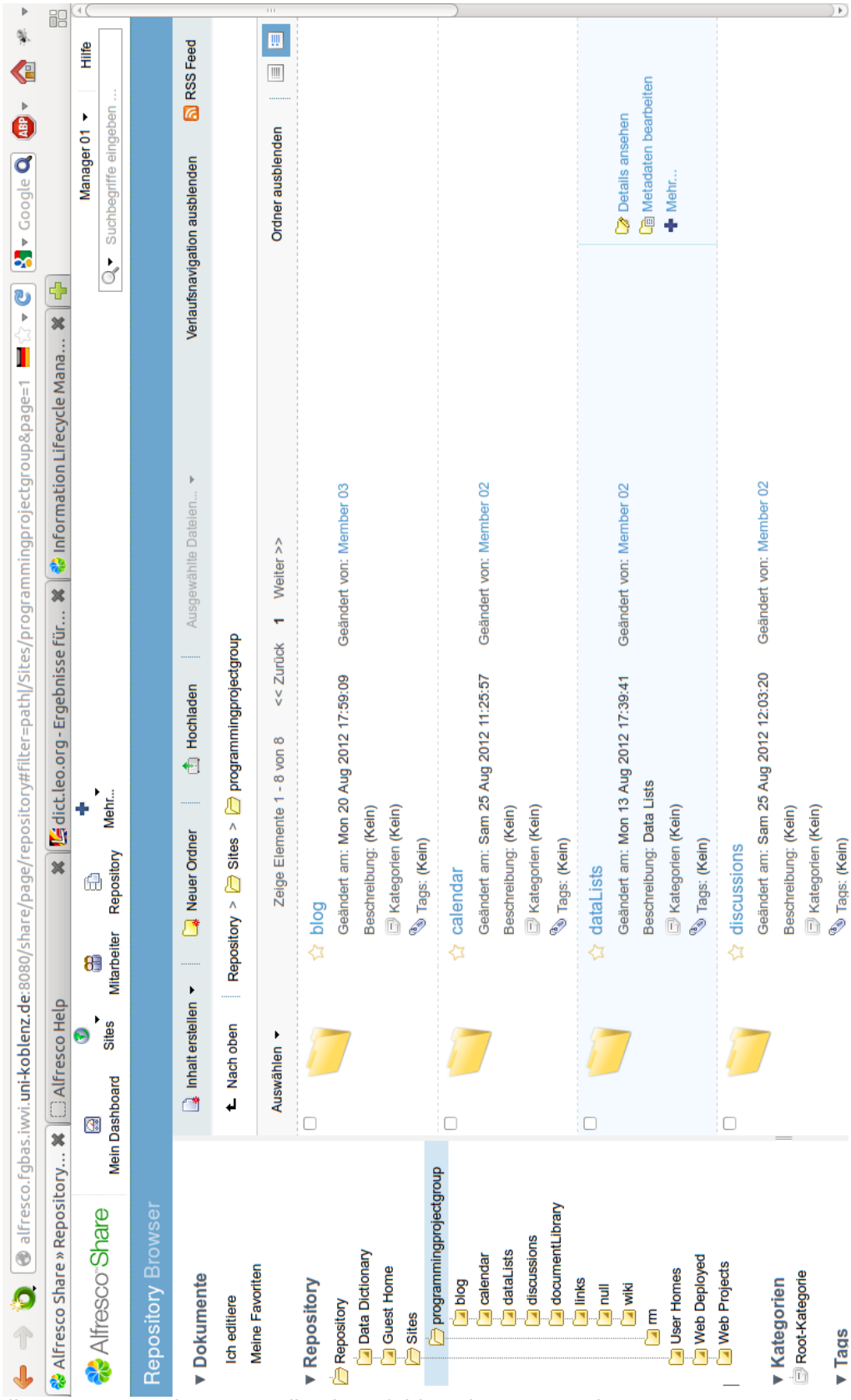
Appendix A-12: Discussion main page (own source)



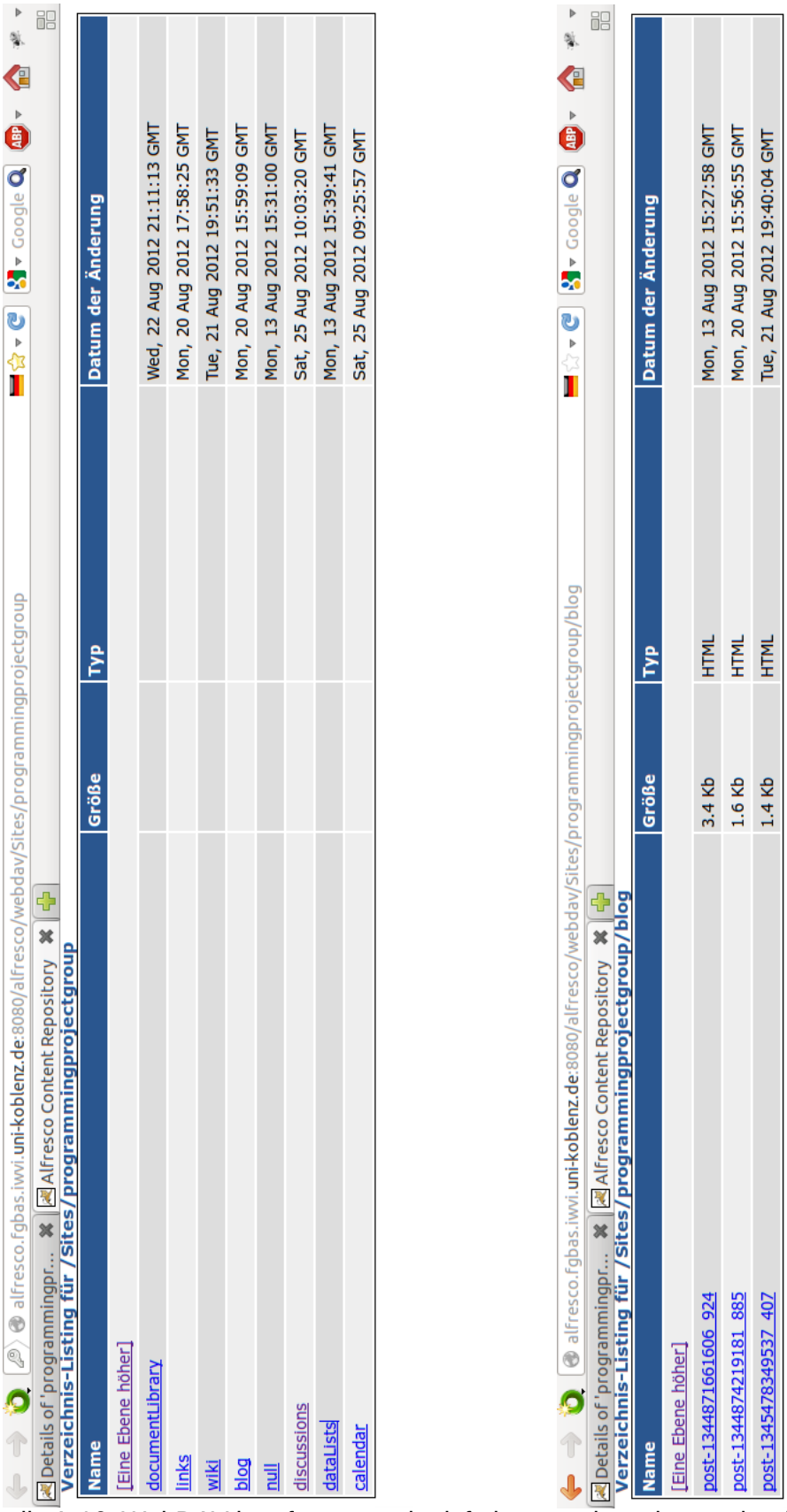
Appendix A-13: Exemplary discussion with answers (own source)



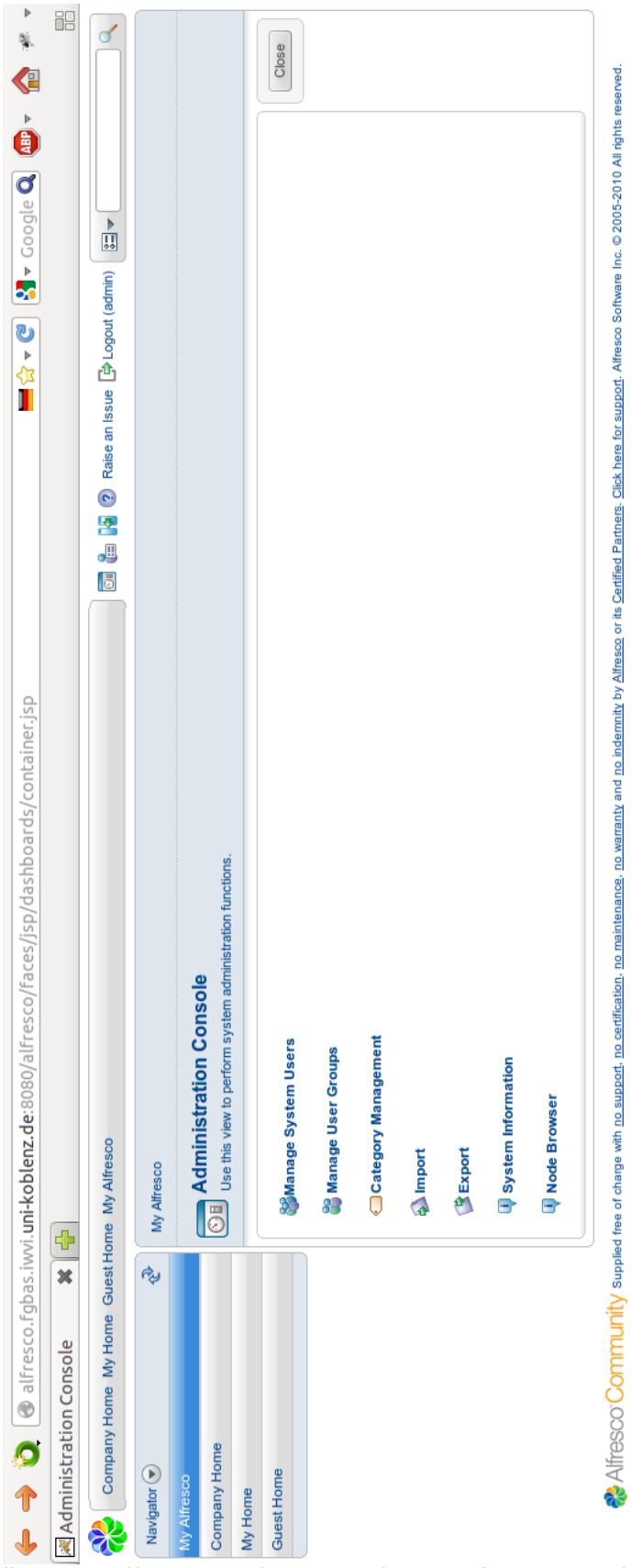
Appendix A-14: Repository main page (own source)



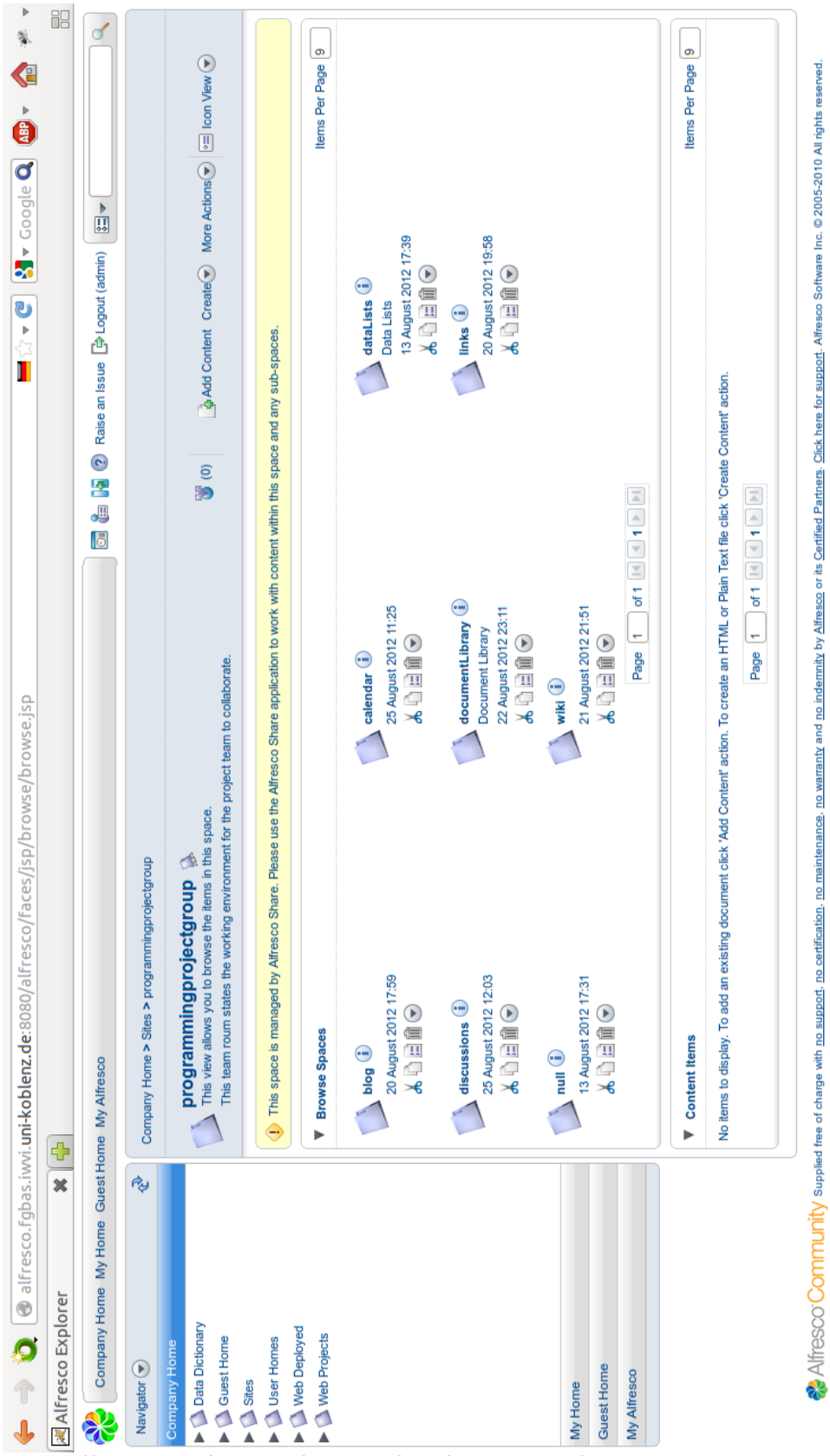
Appendix A-15: Repository – applications folders (own source)



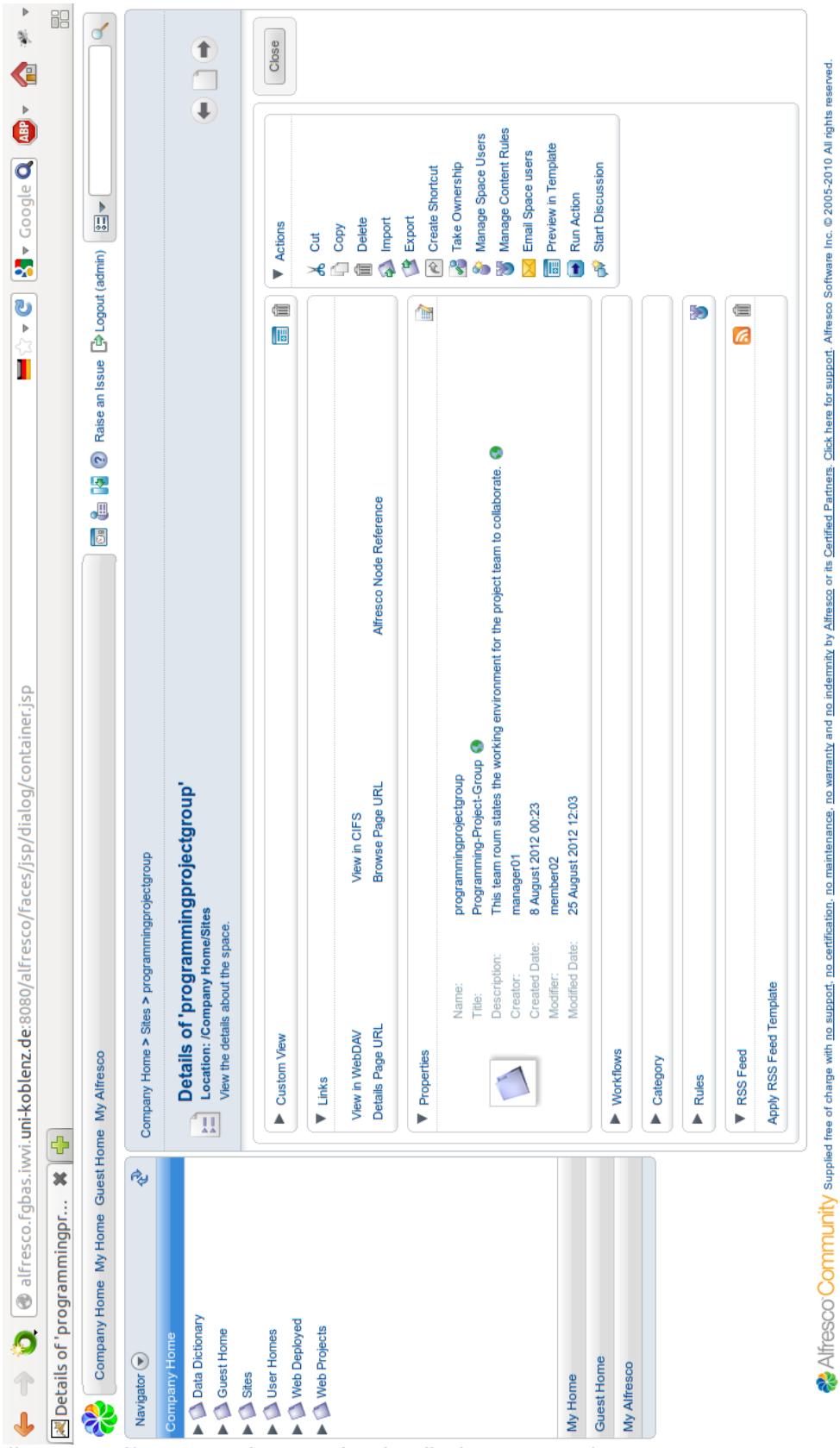
Appendix A-16: WebDAV interface – on the left the complete site, on the right the folder "blog" (own source)



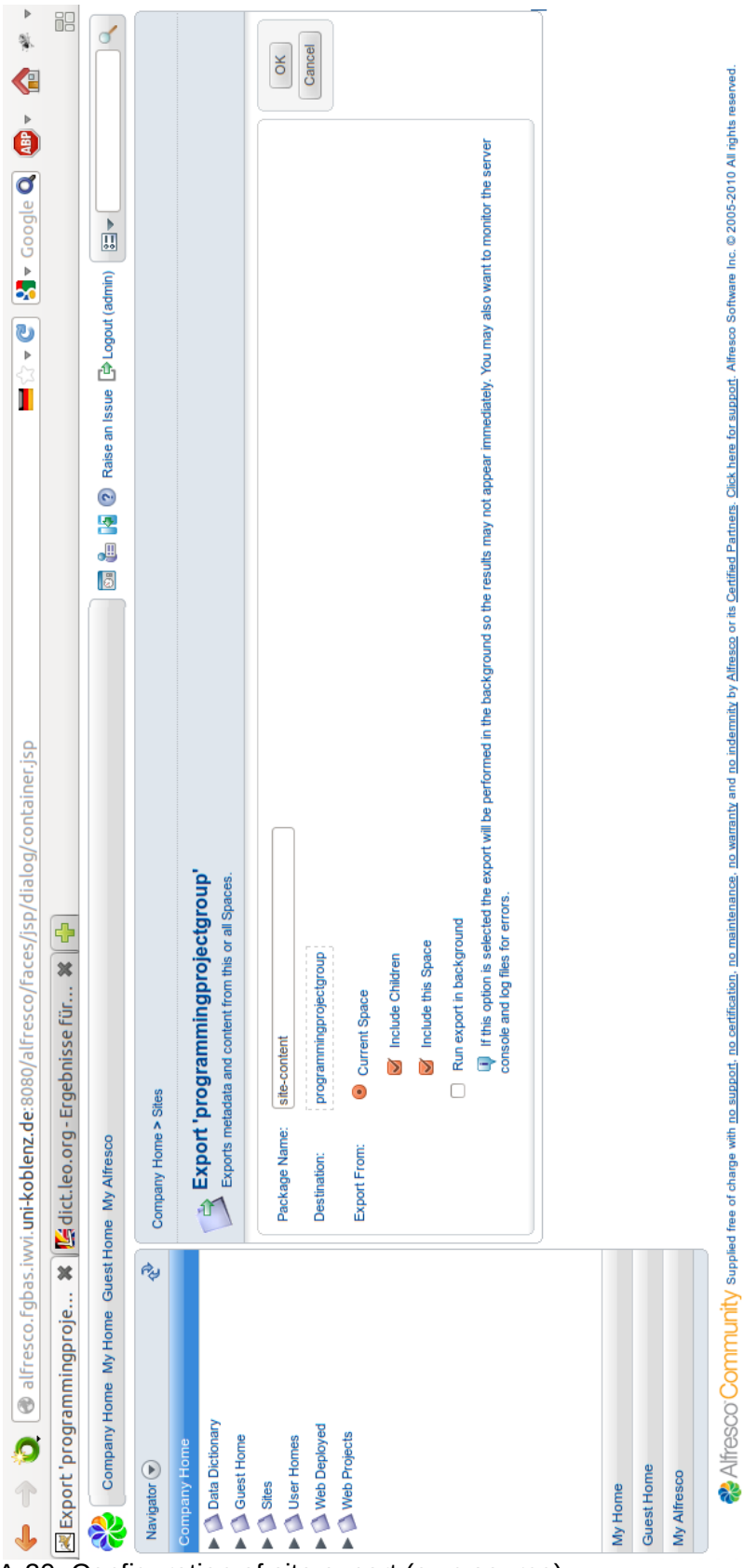
Appendix A-17: Alfresco Explorer – main page (own source)



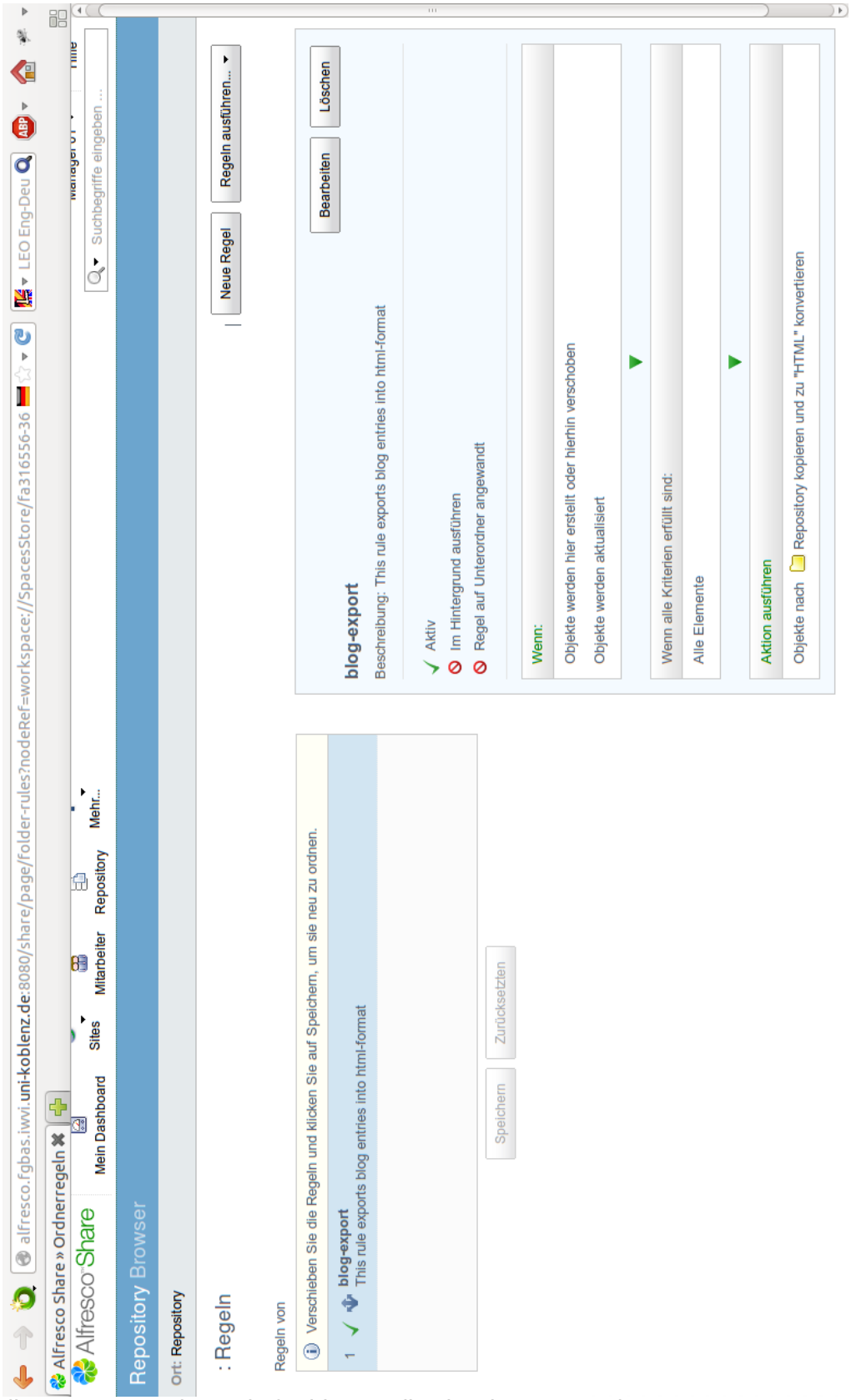
Appendix A-18: Alfresco Explorer – site overview (own source)



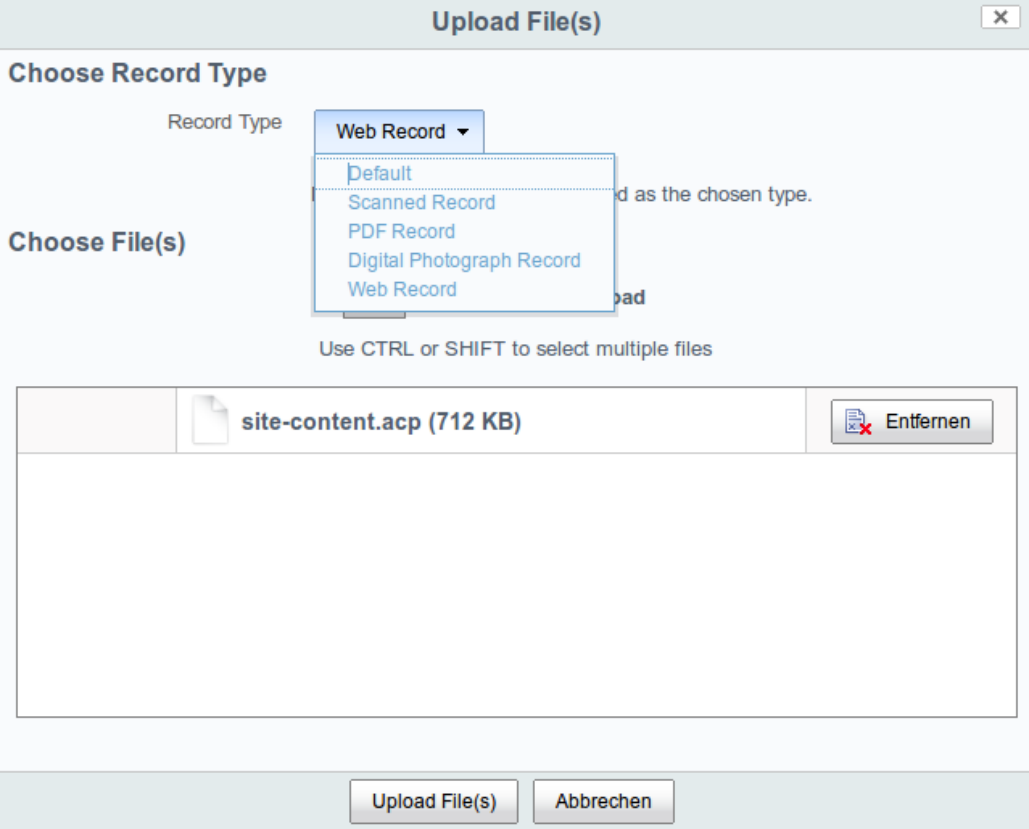
Appendix A-19: Alfresco Explorer – site details (own source)



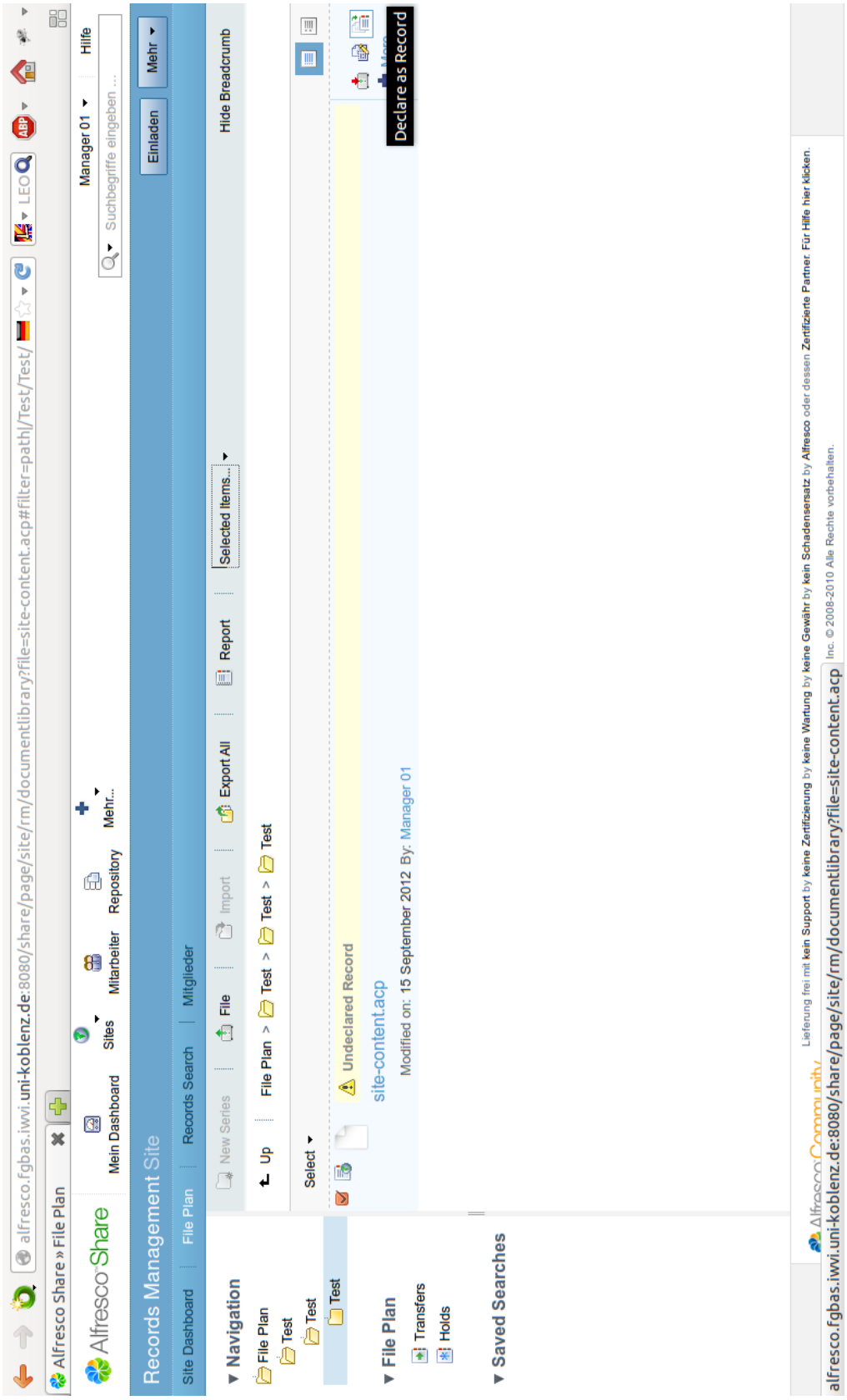
Appendix A-20: Configuration of site export (own source)



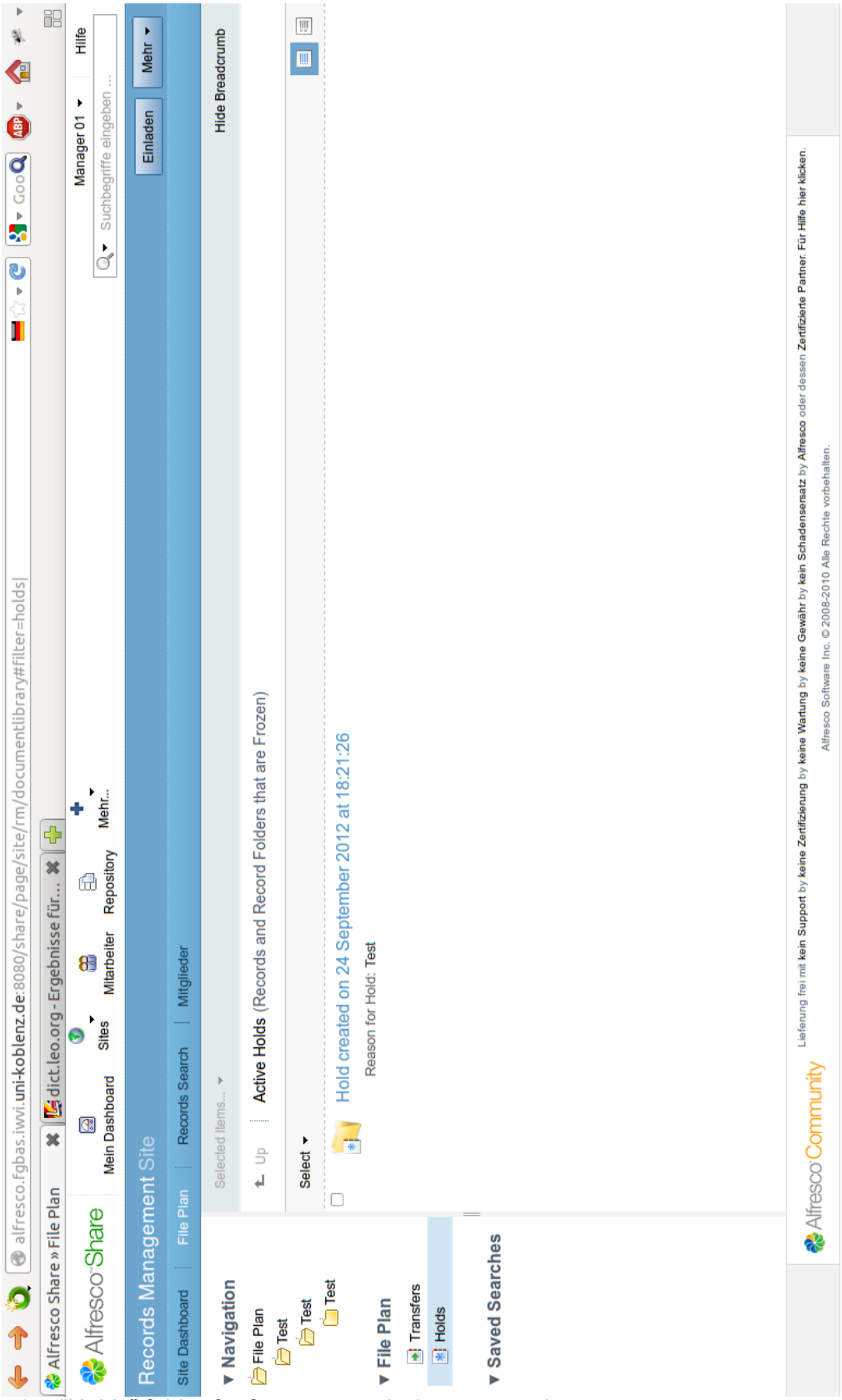
Appendix A-21: Exemplary rule for blog application (own source)



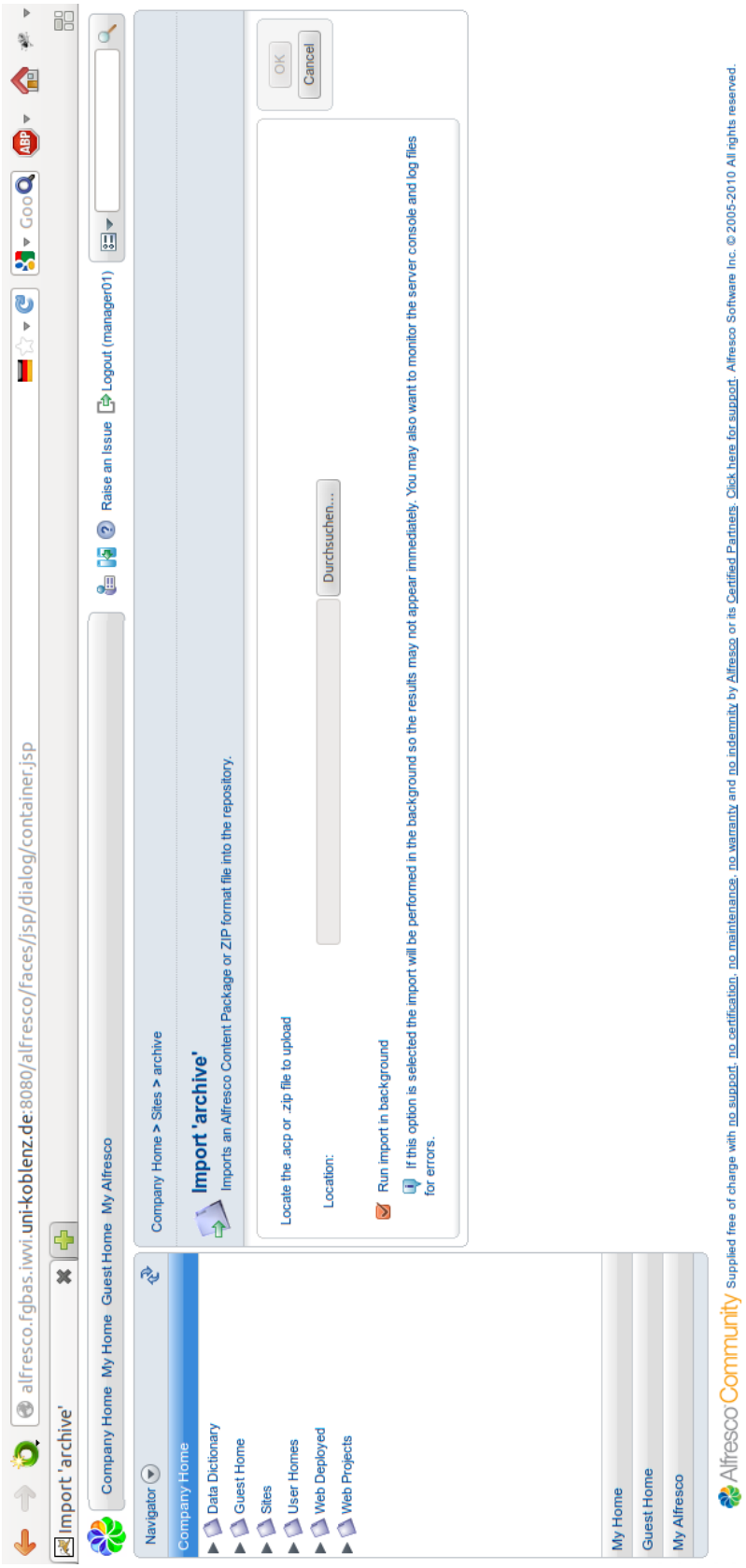
Appendix A-22: Record types offered by Alfresco (own source)



Appendix A-23: Declaring an acp-file as record (own source)

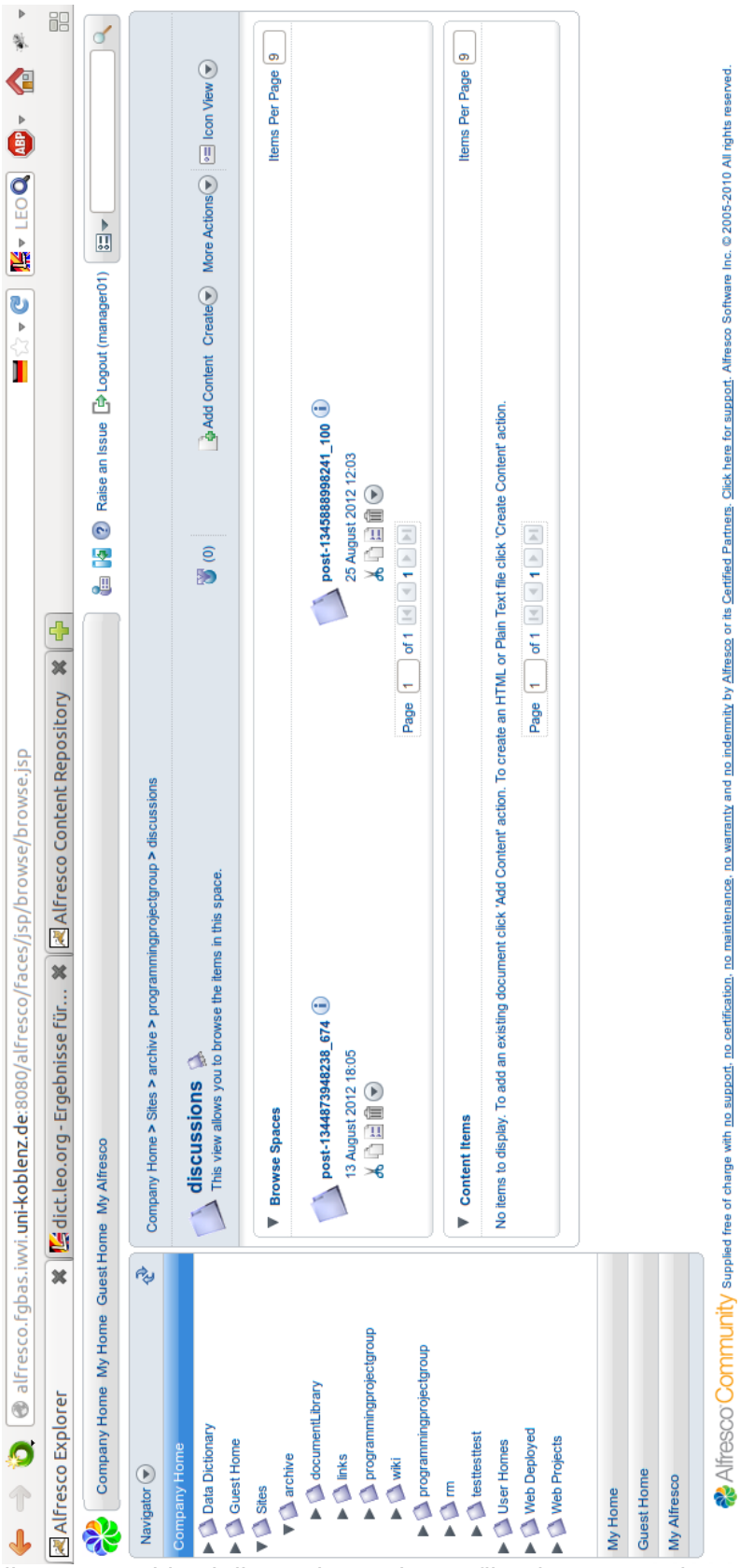


Appendix A-24: “Holds” folder for frozen records (own source)



Appendix A-25: Import dialog for archived sites (own source)

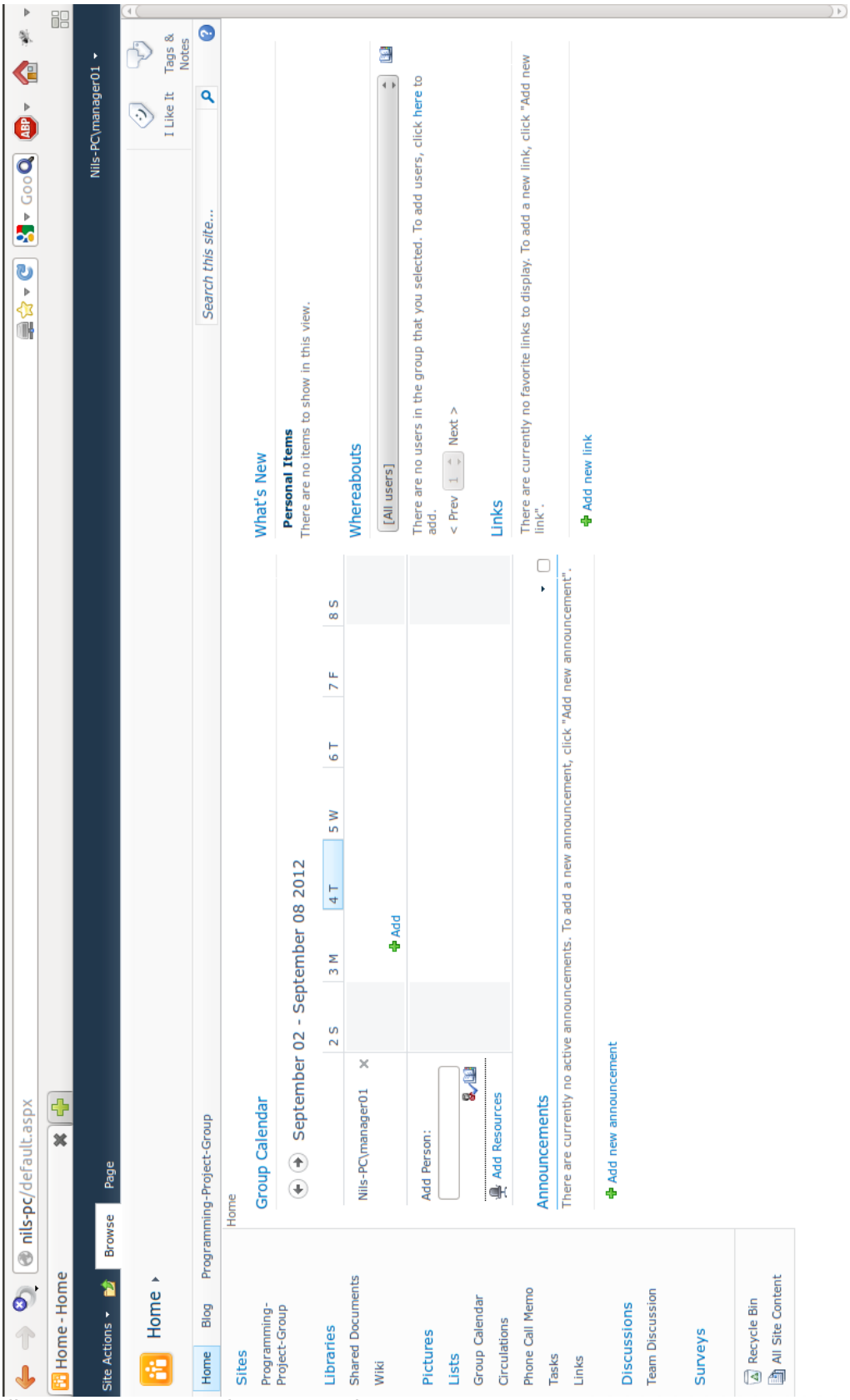
Appendix A-26: Imported site displayed as a folder (own source)



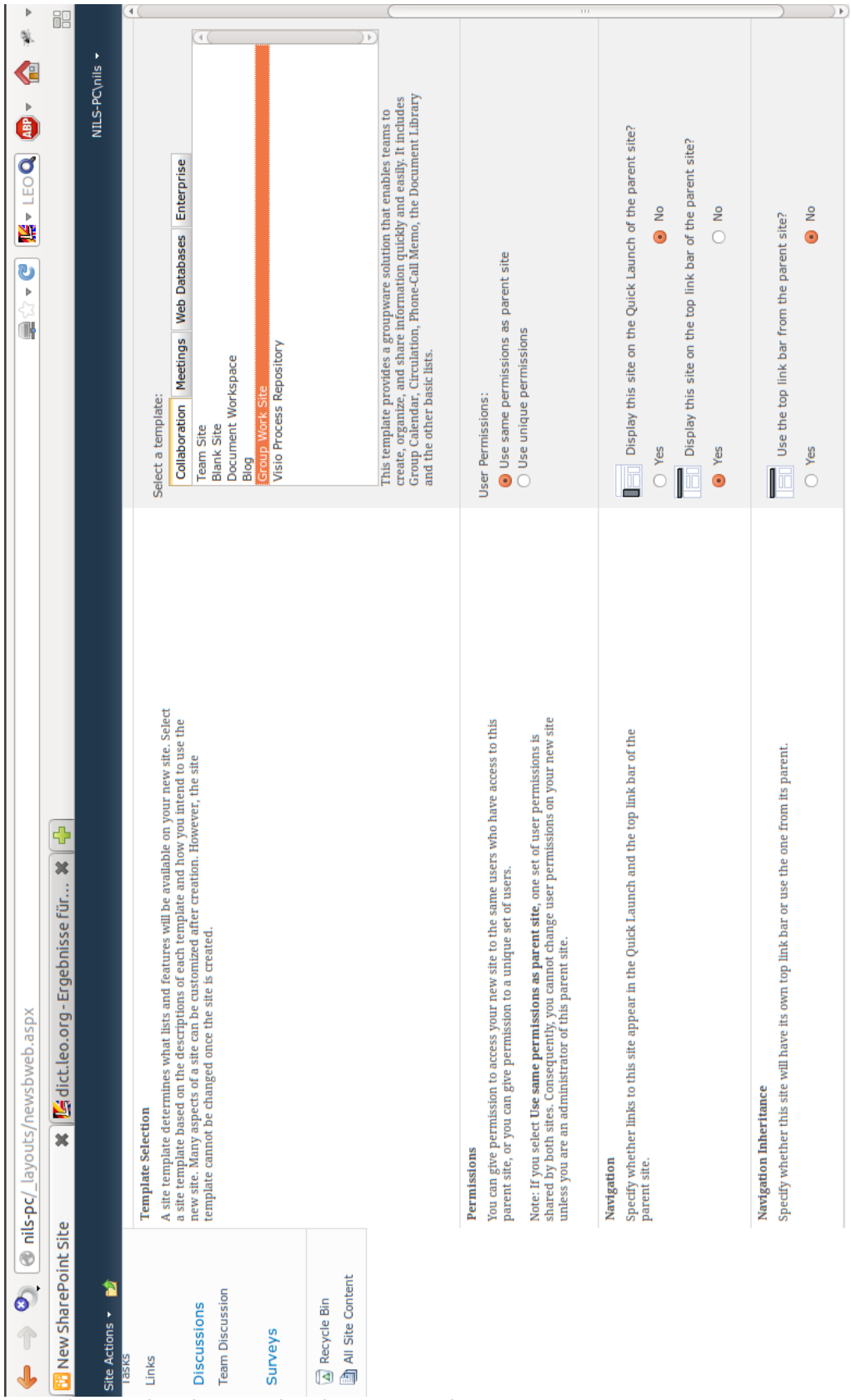
Appendix A-27: Archived discussion topics as files (own source)

Appendix B – SharePoint

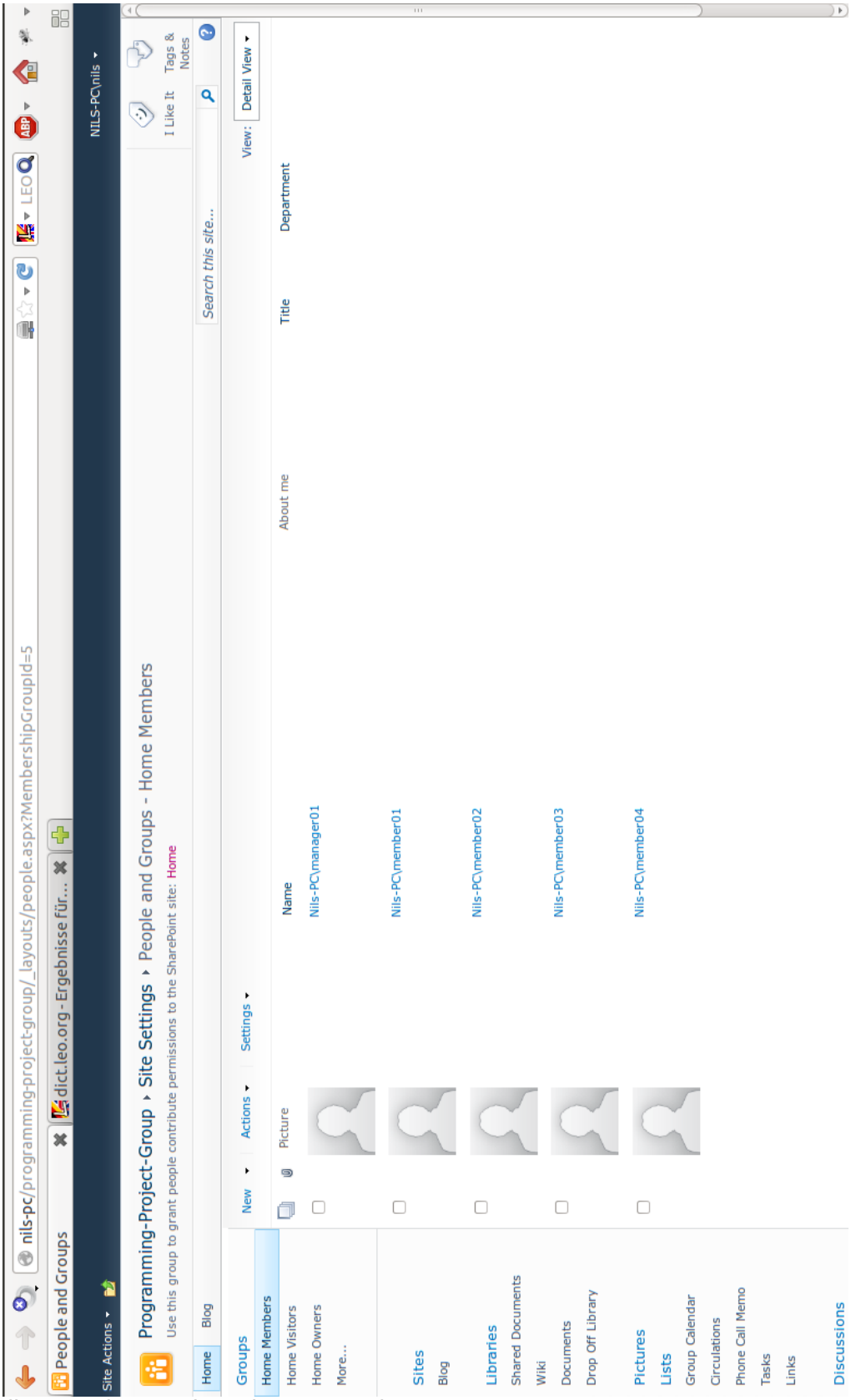
Appendix B-1: SharePoint features listed (own source)



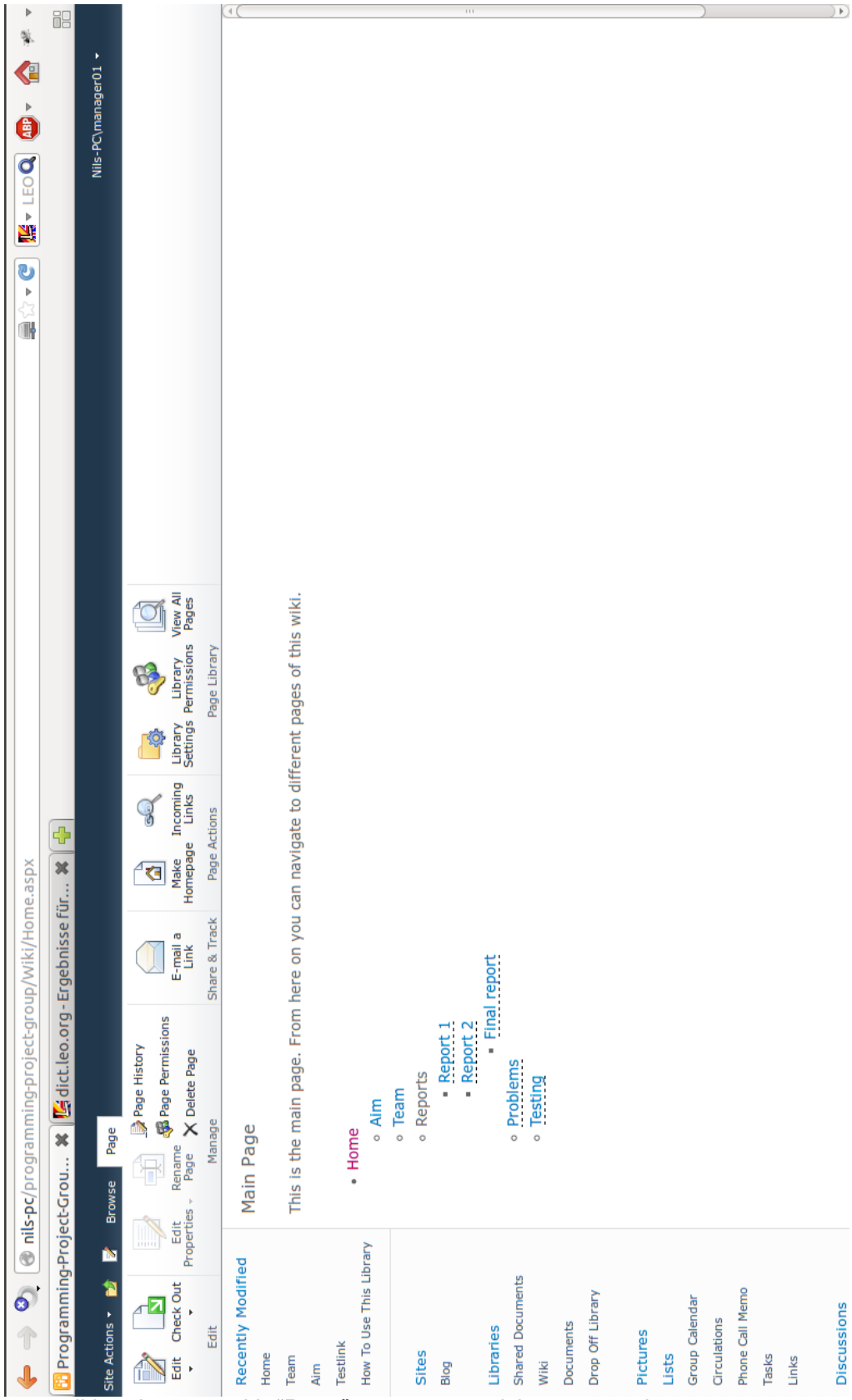
Appendix B-2: Home page (own source)



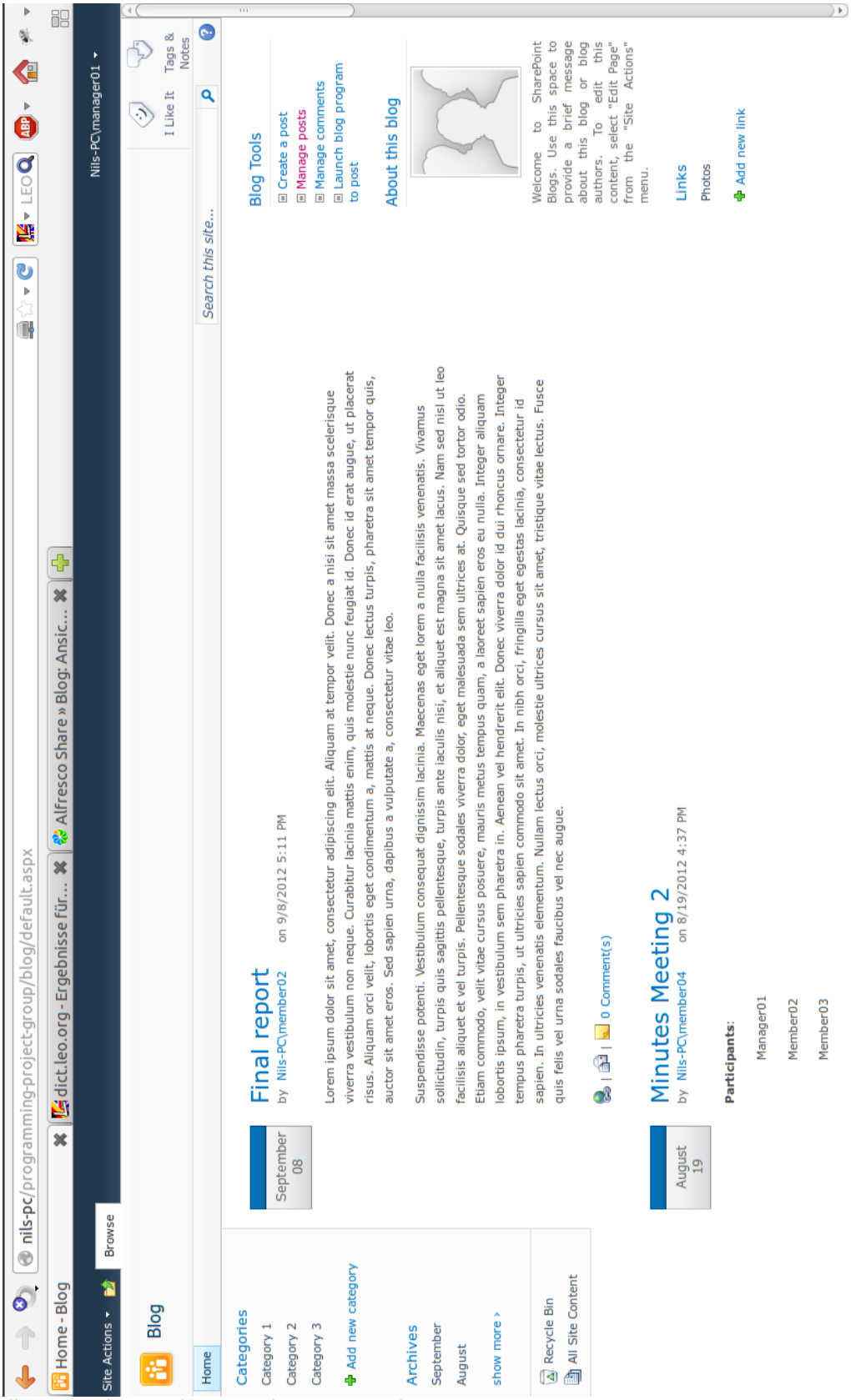
Appendix B-3: SharePoint site creation (own source)



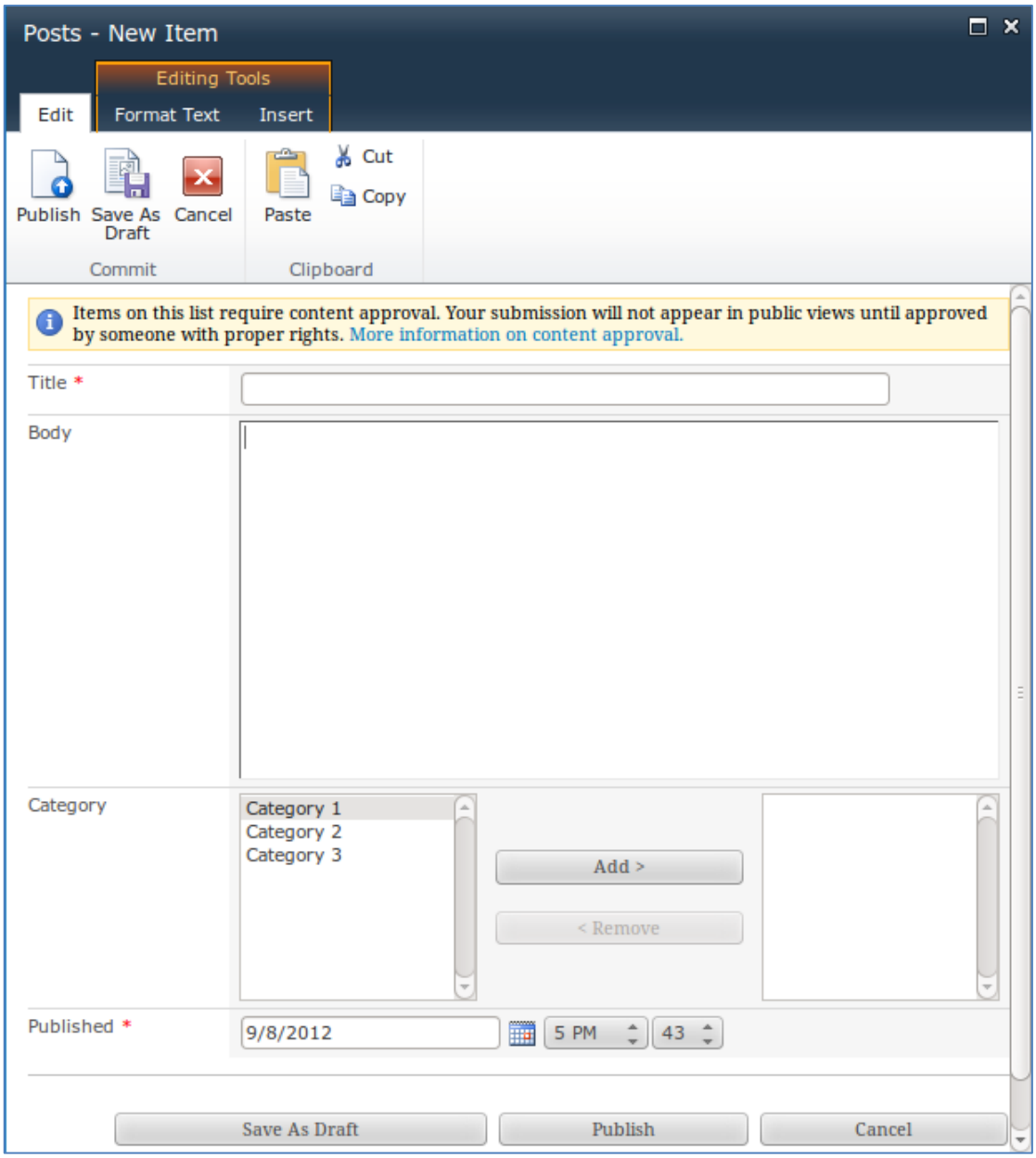
Appendix B-4: Site members (own source)



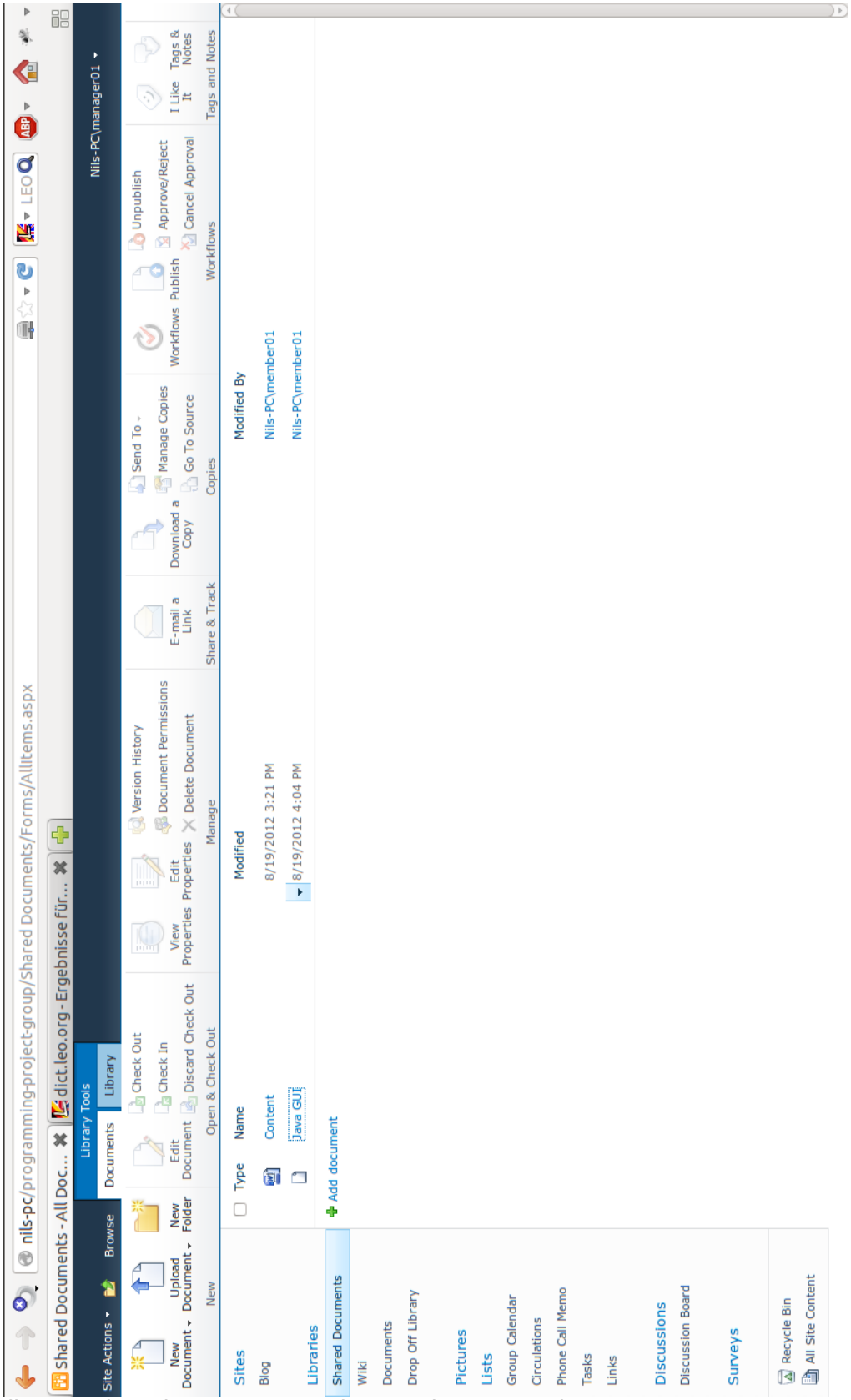
Appendix B-5: Wiki main page with “Page” menu opened (own source)



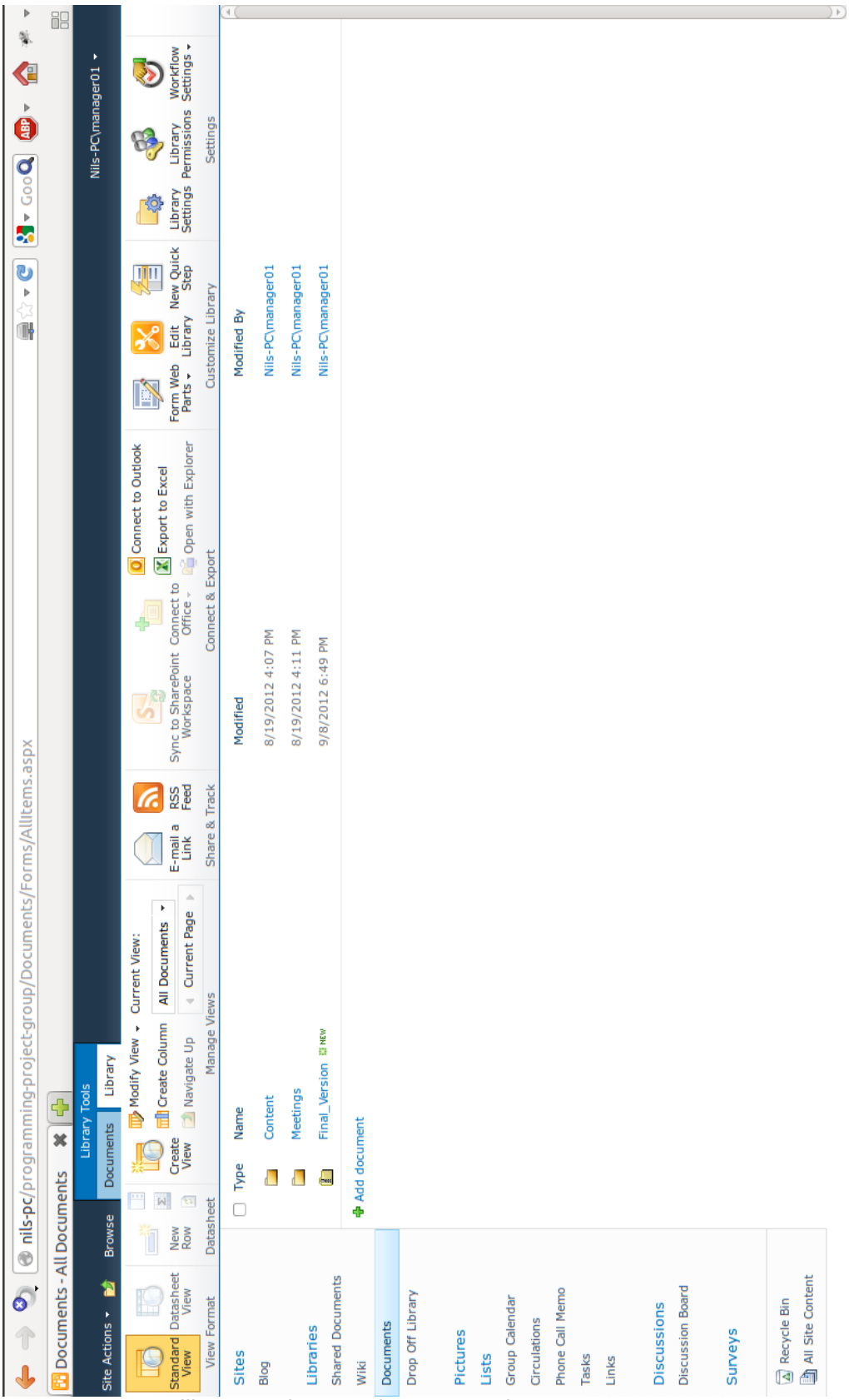
Appendix B-6: Blog main page (own source)



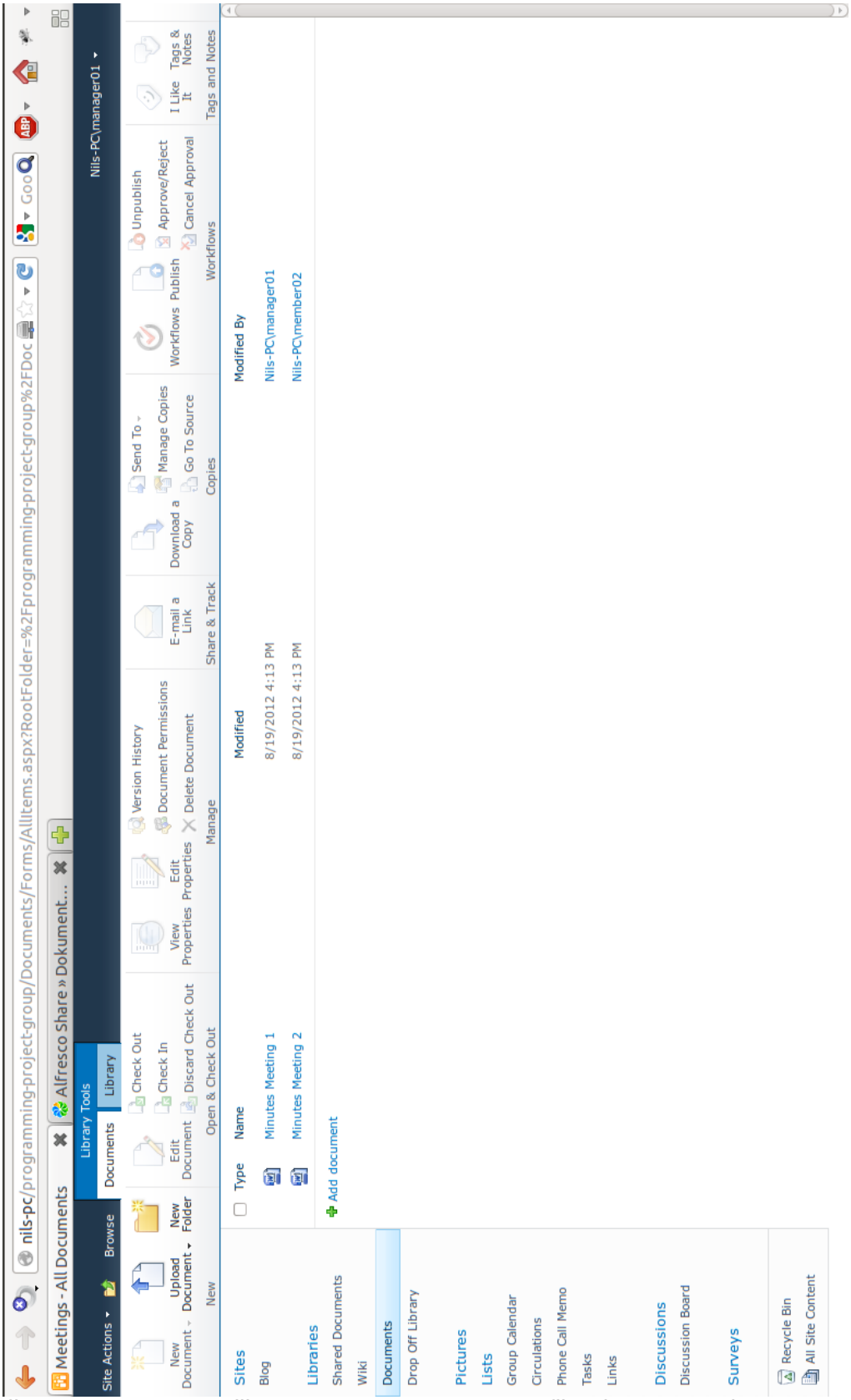
Appendix B-7: Blog post creation (own source)



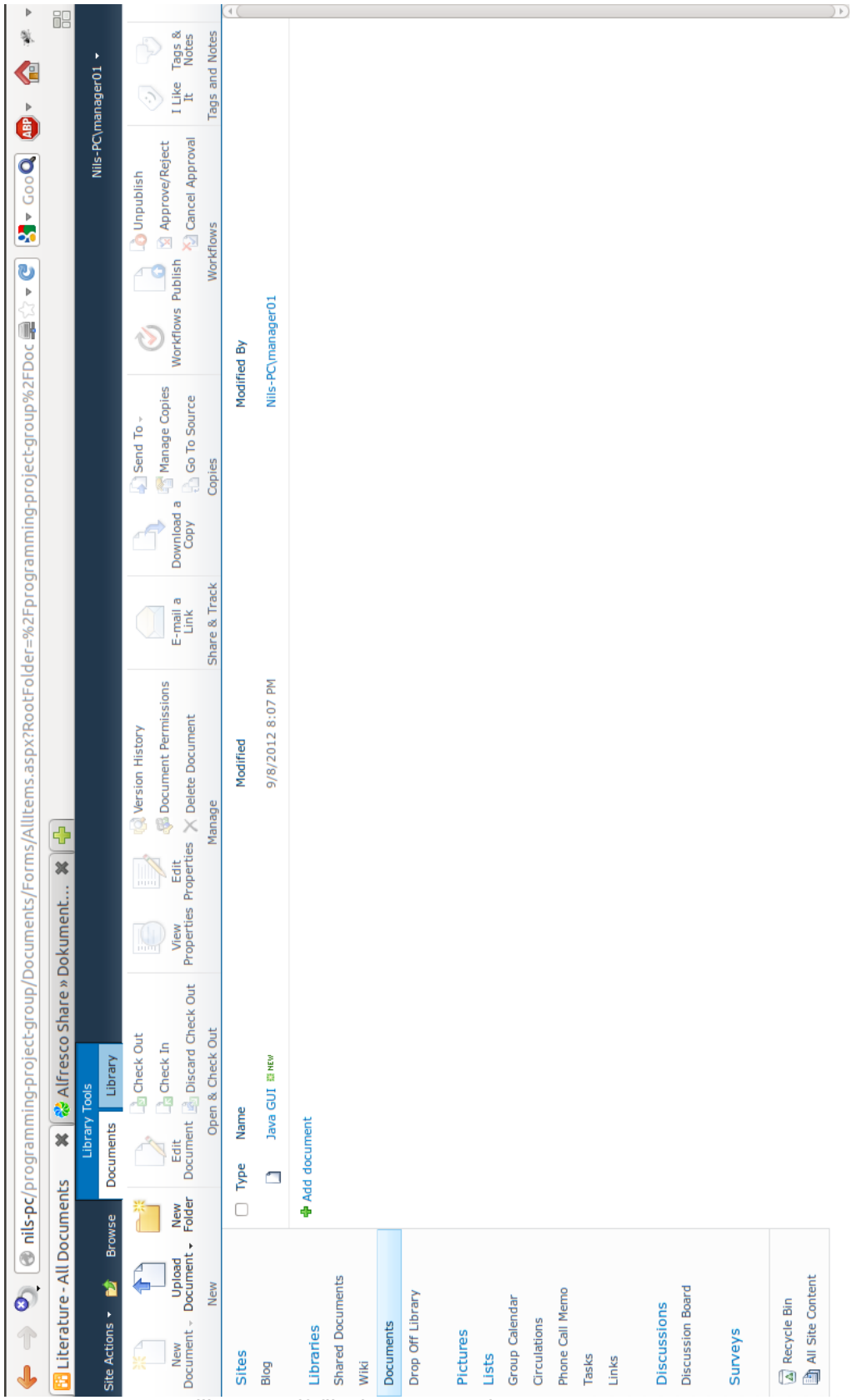
Appendix B-8: Shared Documents main page (own source)



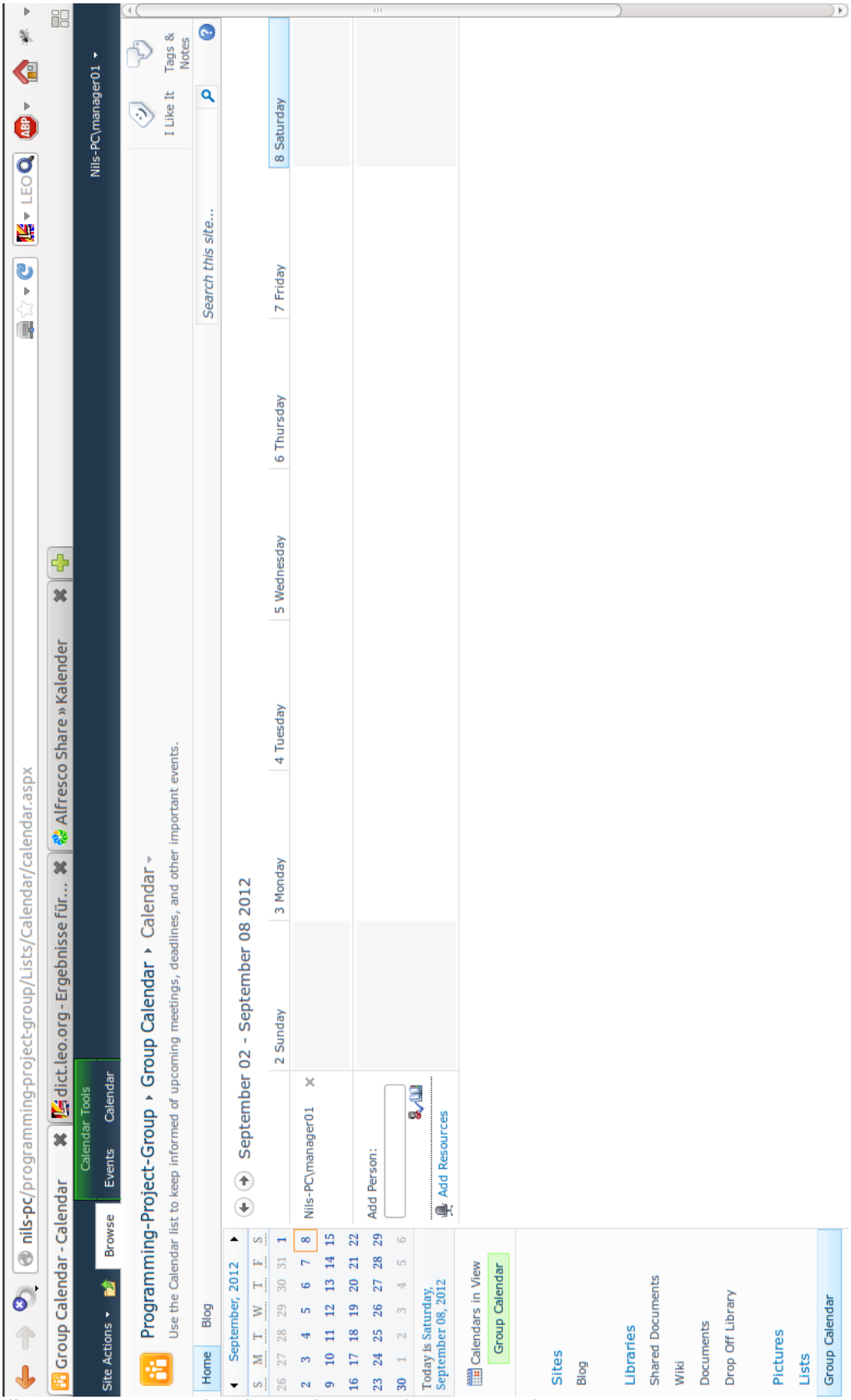
Appendix B-9: Documents library main page (own source)



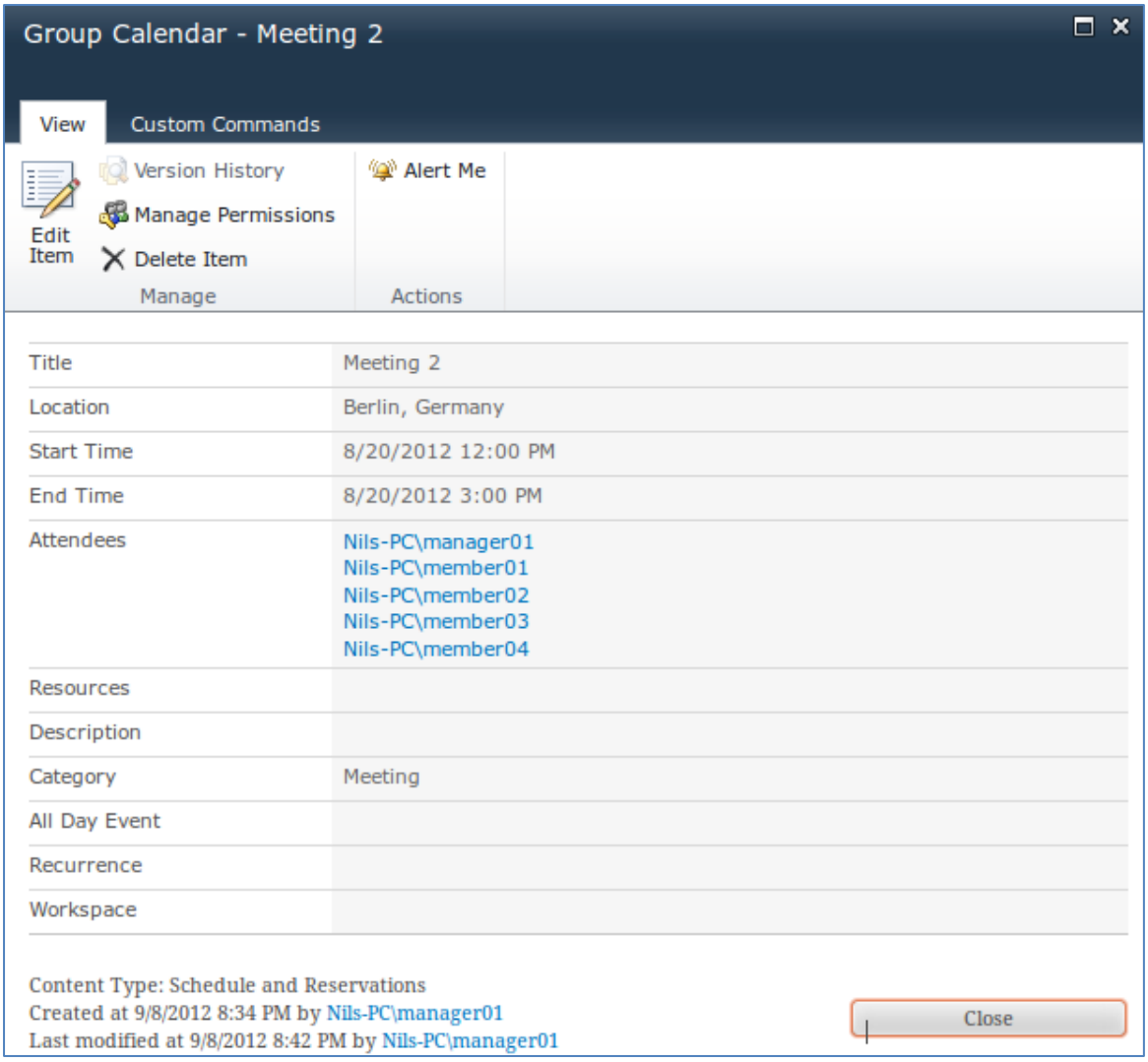
Appendix B-10: Documents library: Open-Document text-files (own source)



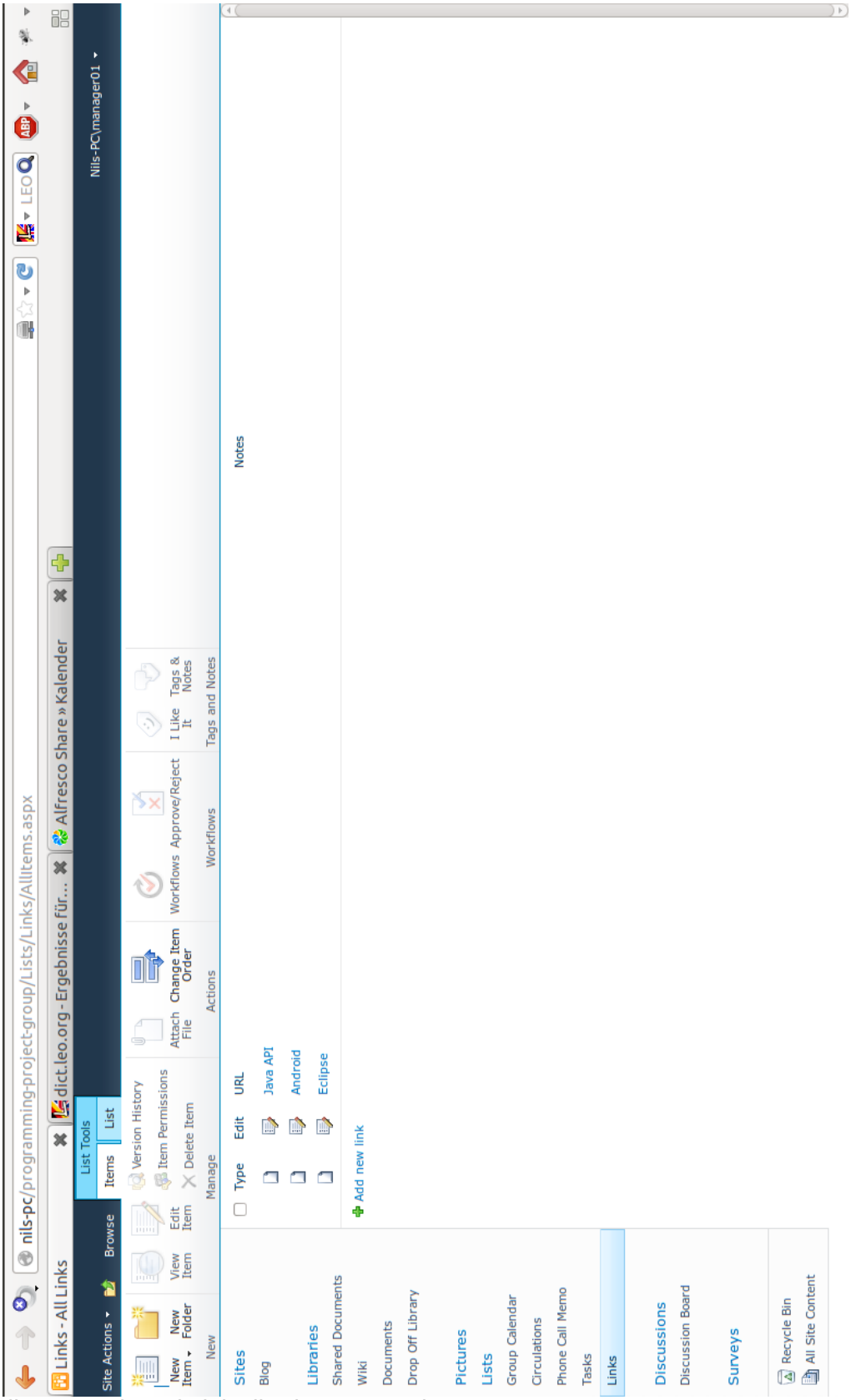
Appendix B-11: Documents library: pdf-file (own source)



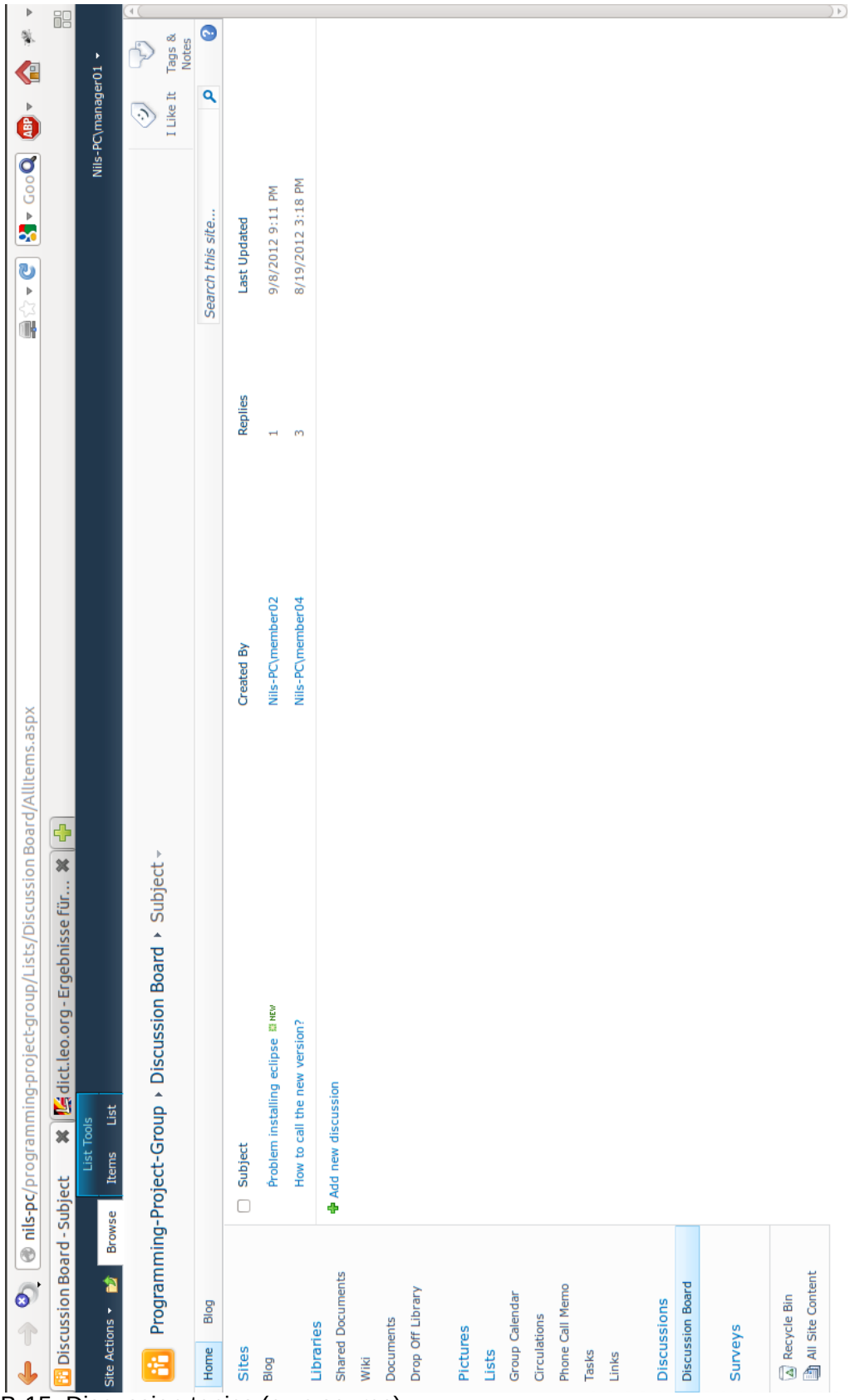
Appendix B-12: Group calendar main page (own source)



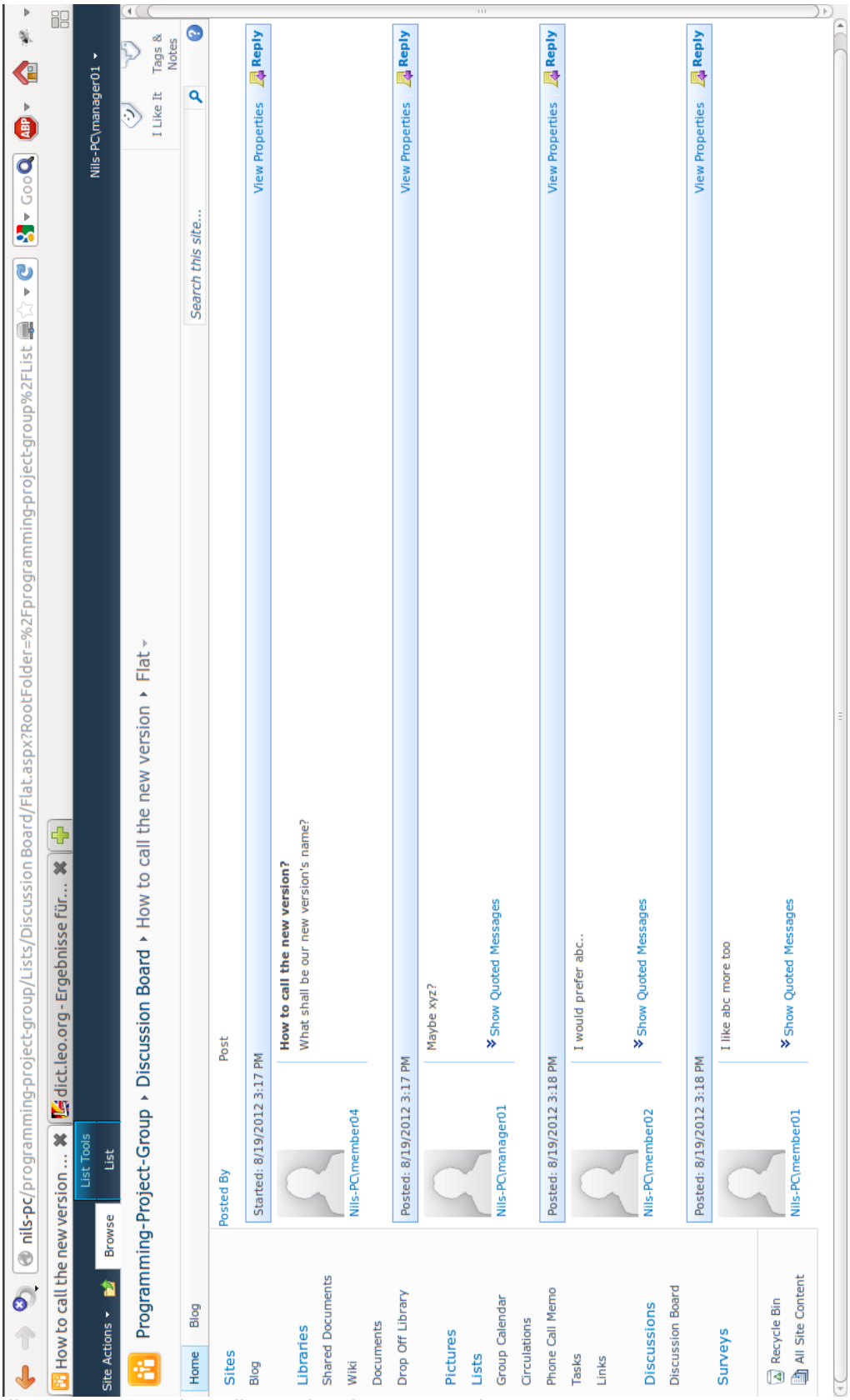
Appendix B-13: Exemplary calendar entry (own source)



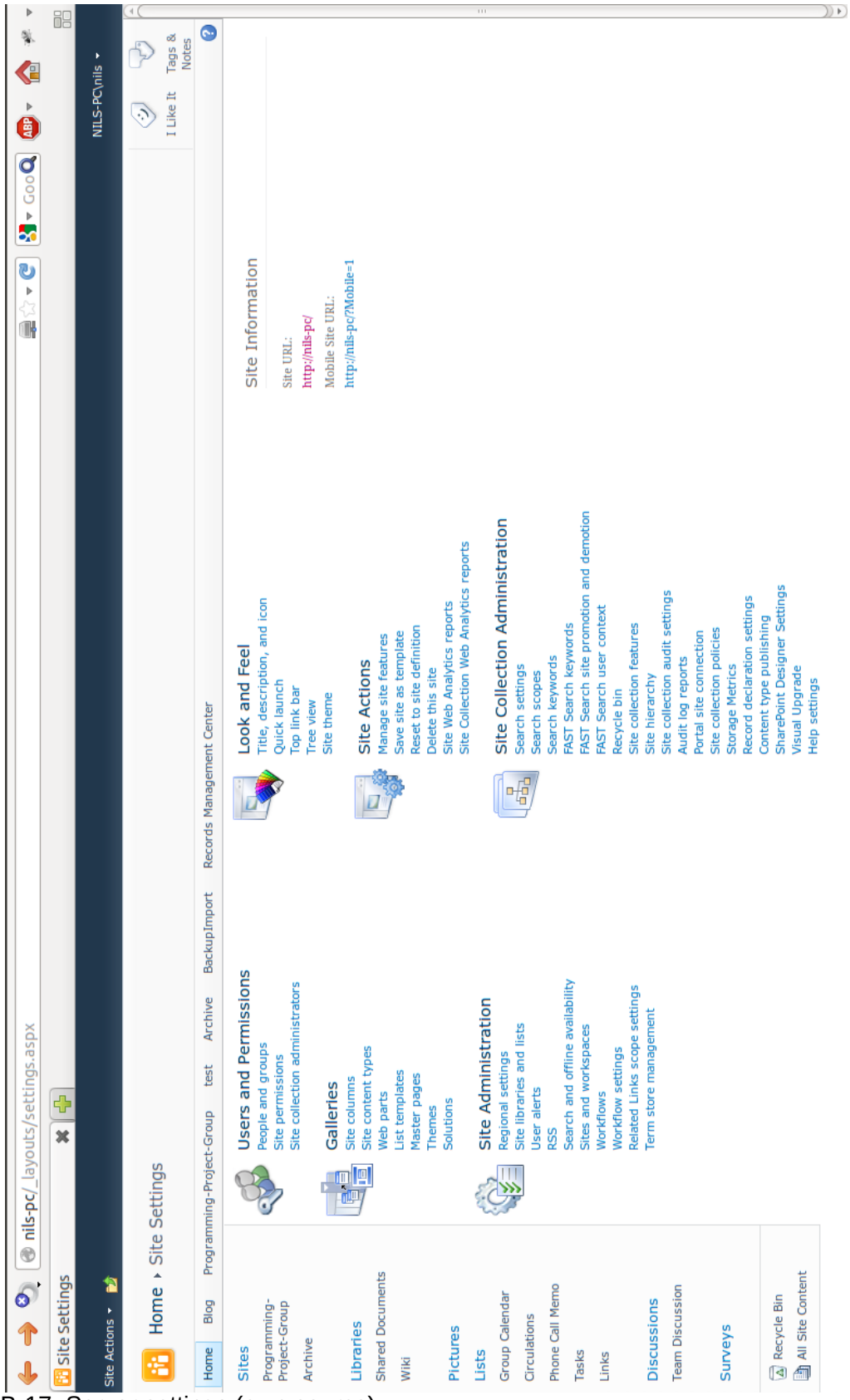
Appendix B-14: Shared Links list (own source)



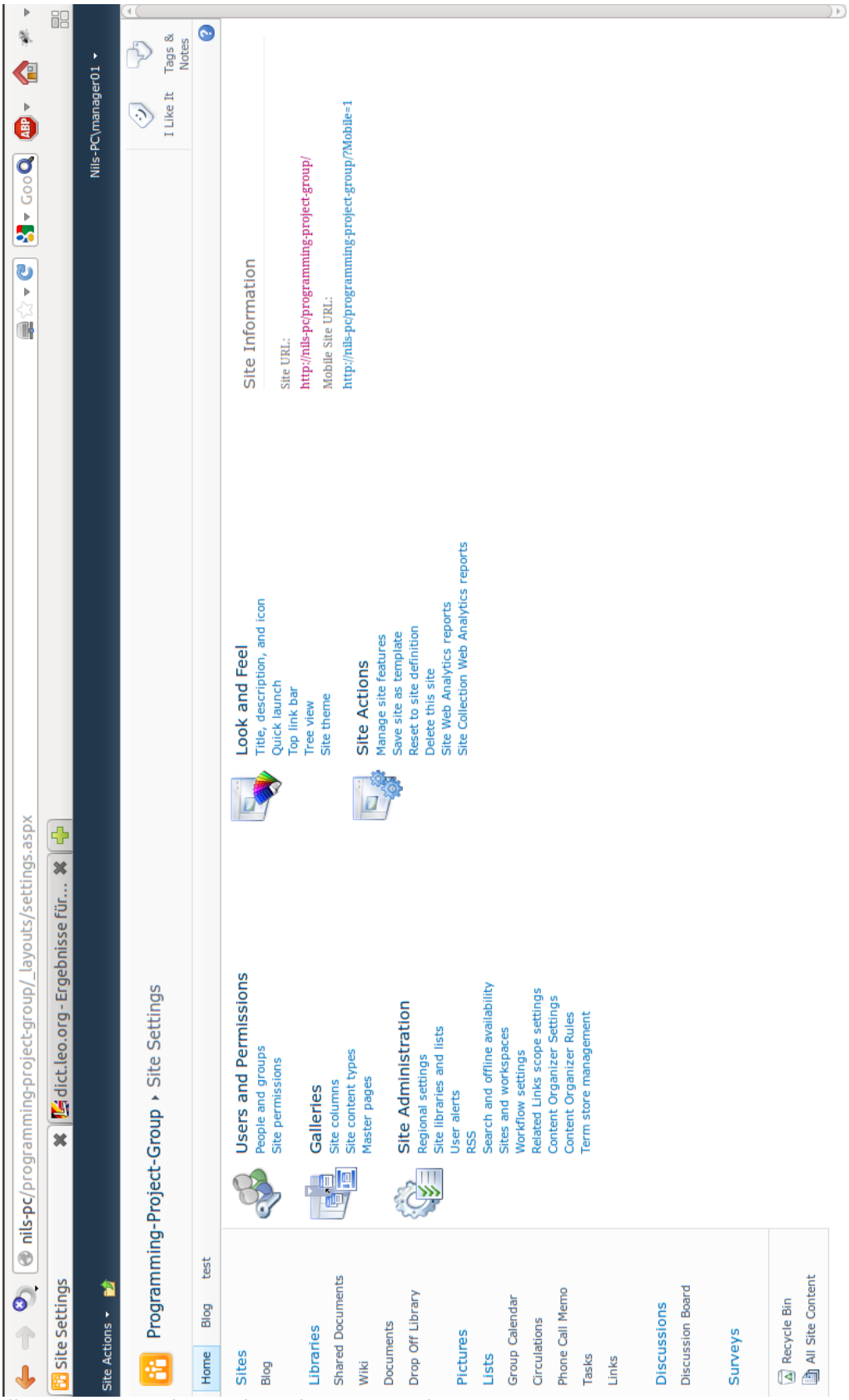
Appendix B-15: Discussion topics (own source)



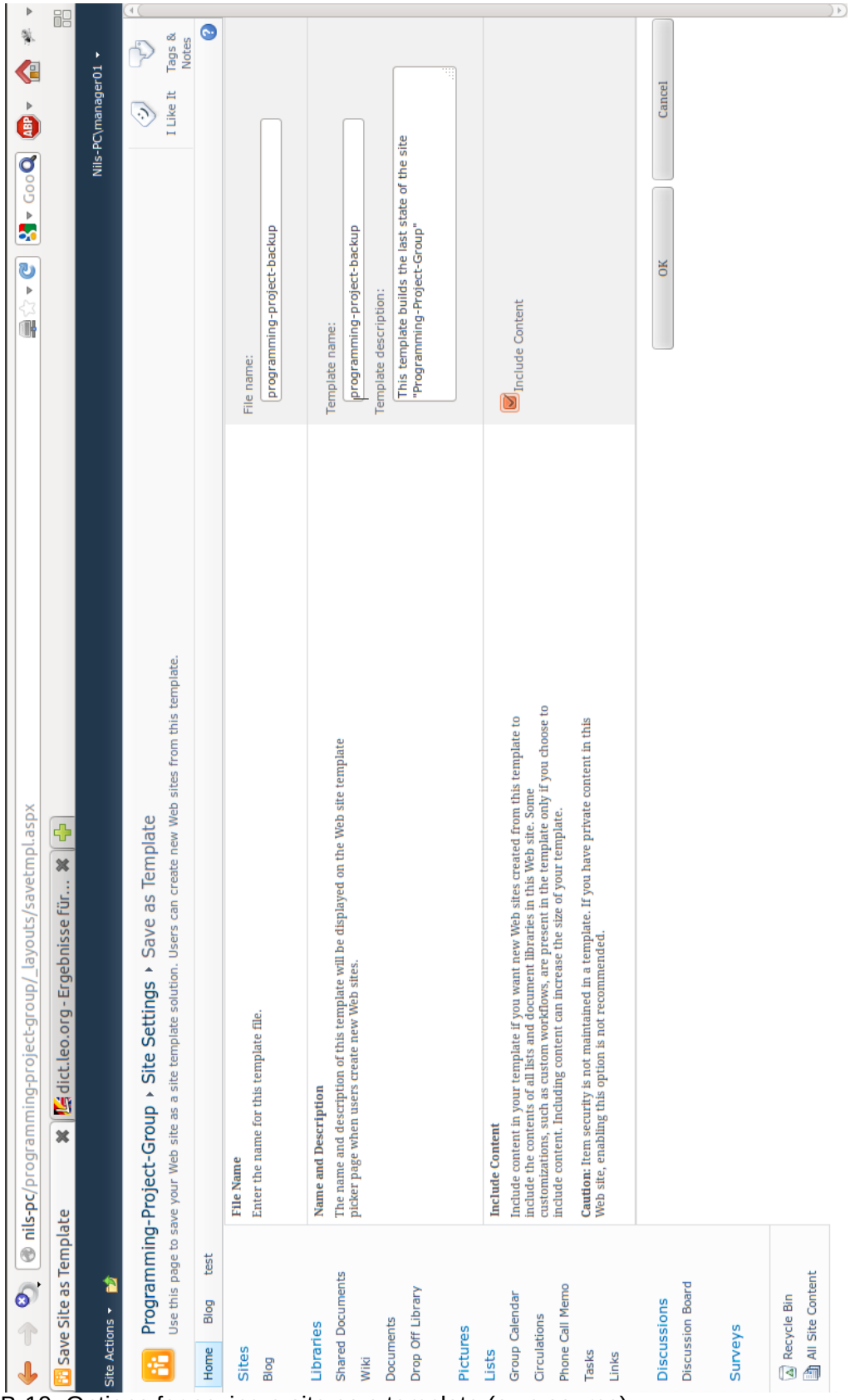
Appendix B-16: Exemplary discussion (own source)



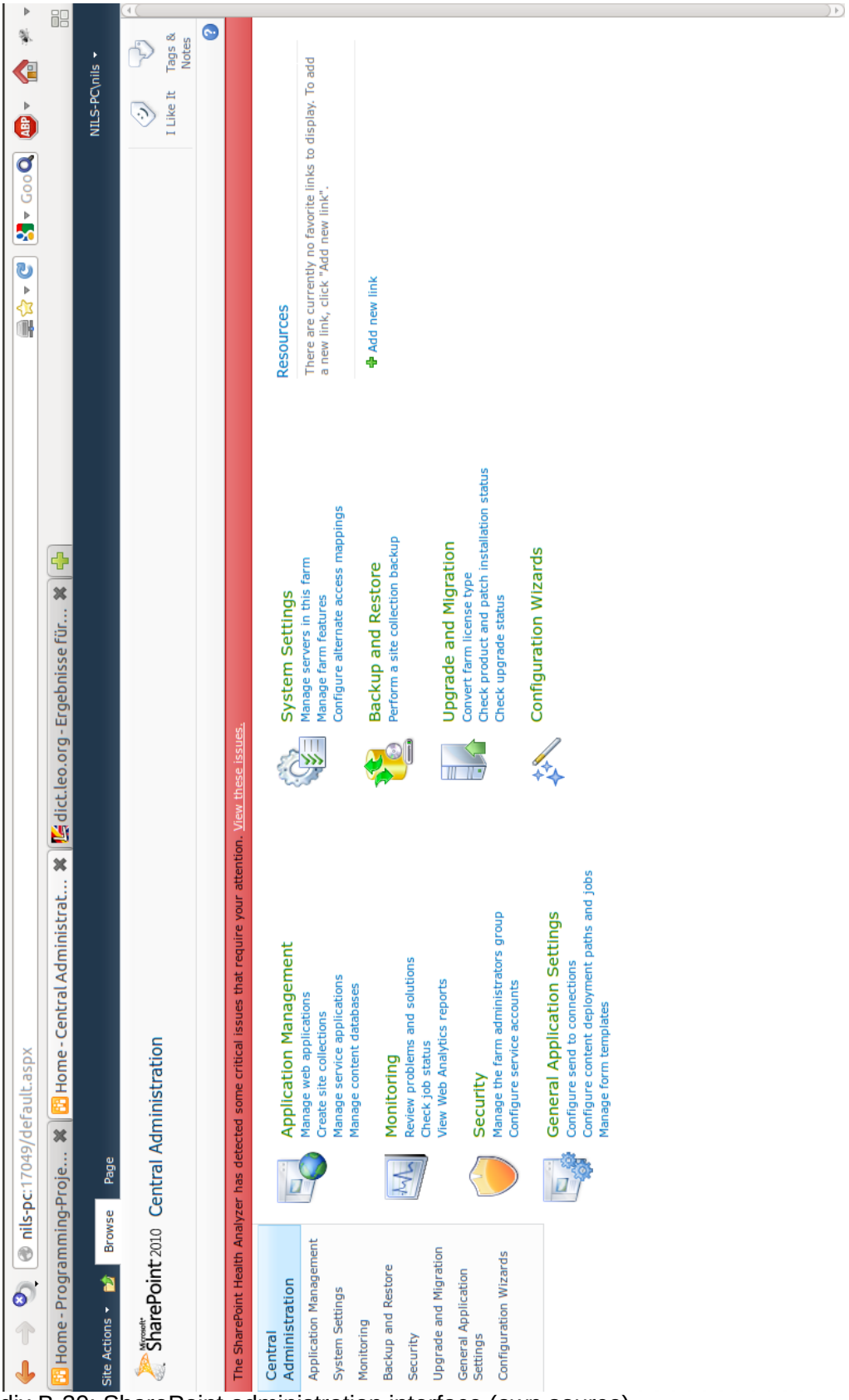
Appendix B-17: Server settings (own source)



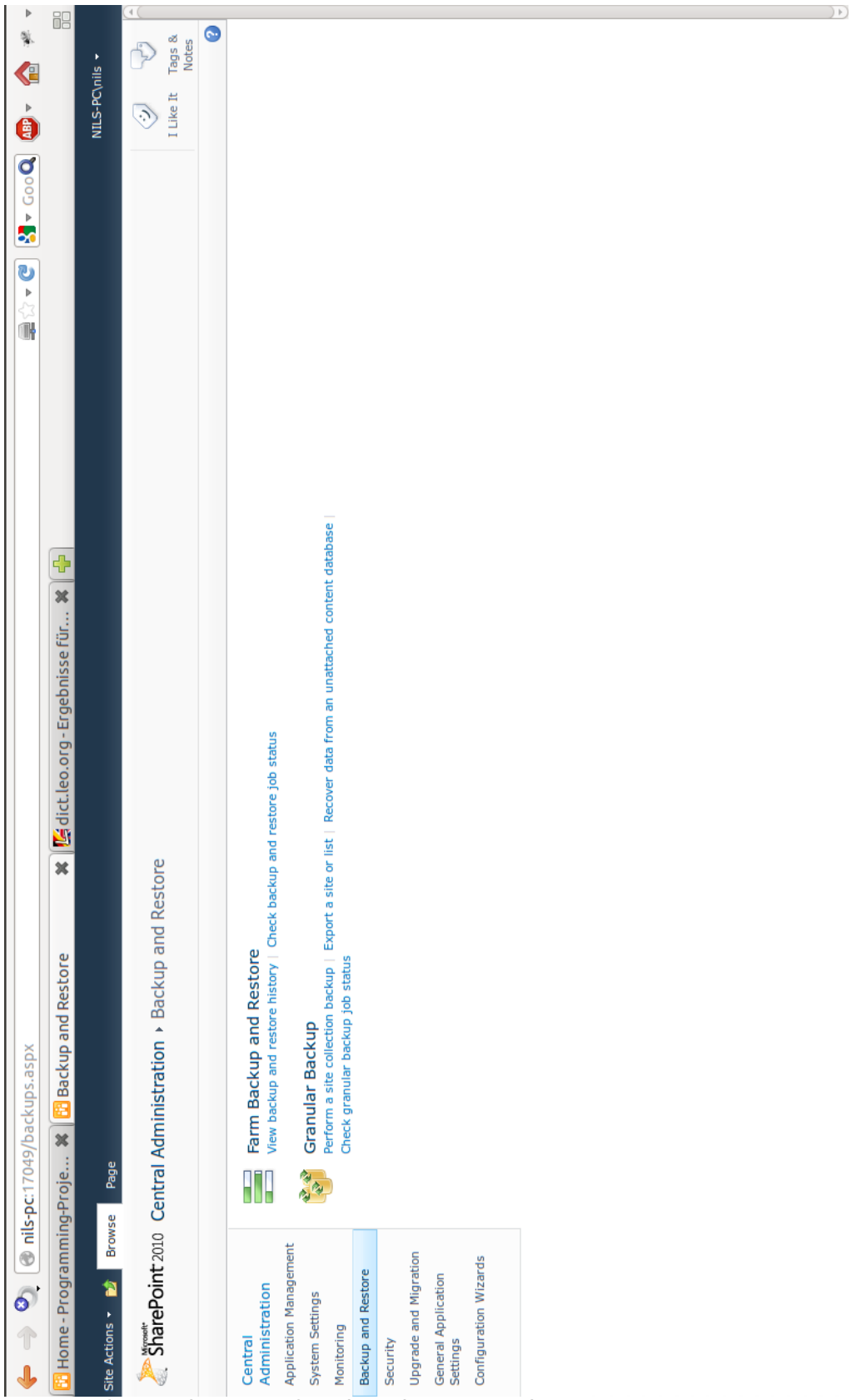
Appendix B-18: Test-site settings (own source)



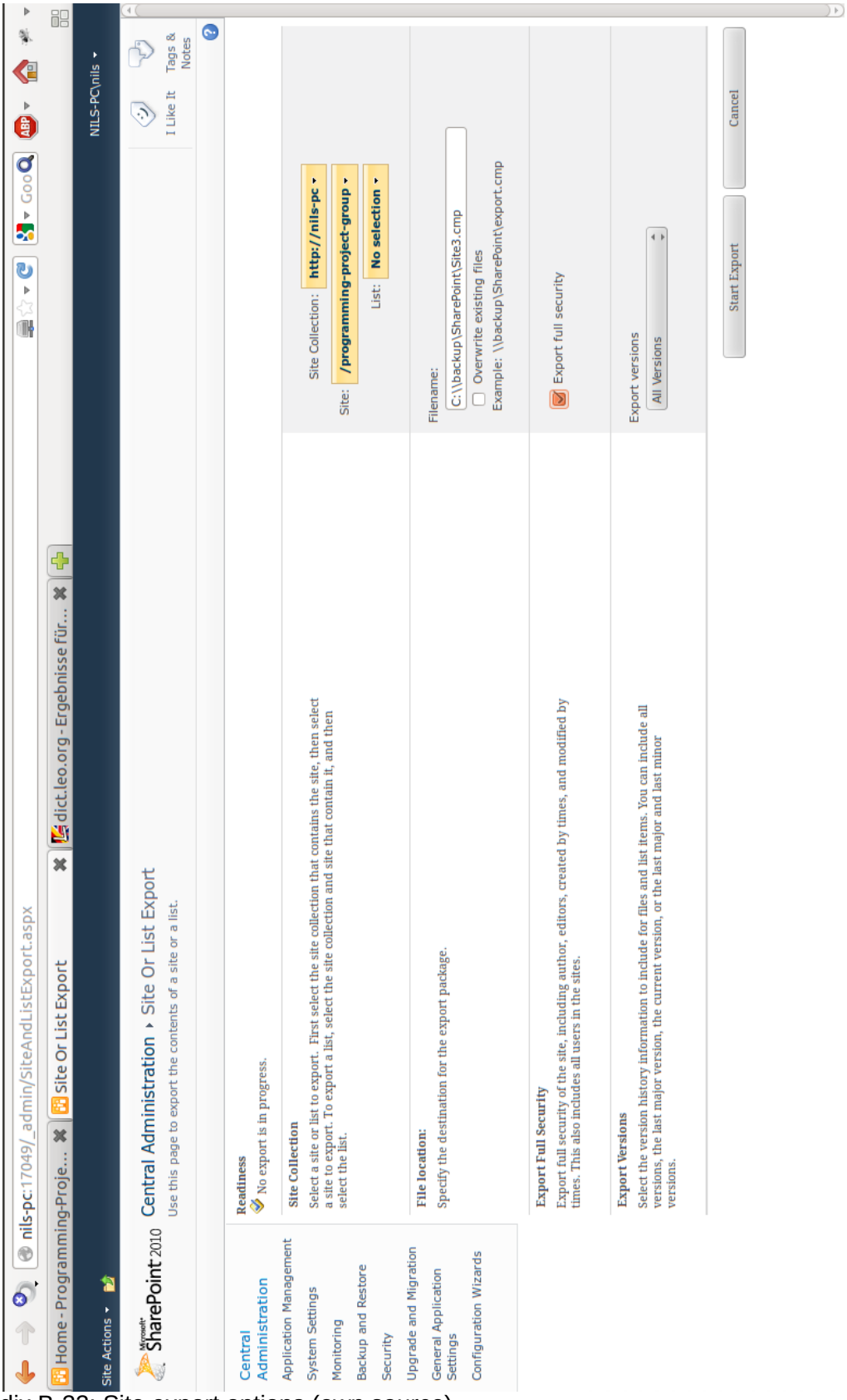
Appendix B-19: Options for saving a site as a template (own source)



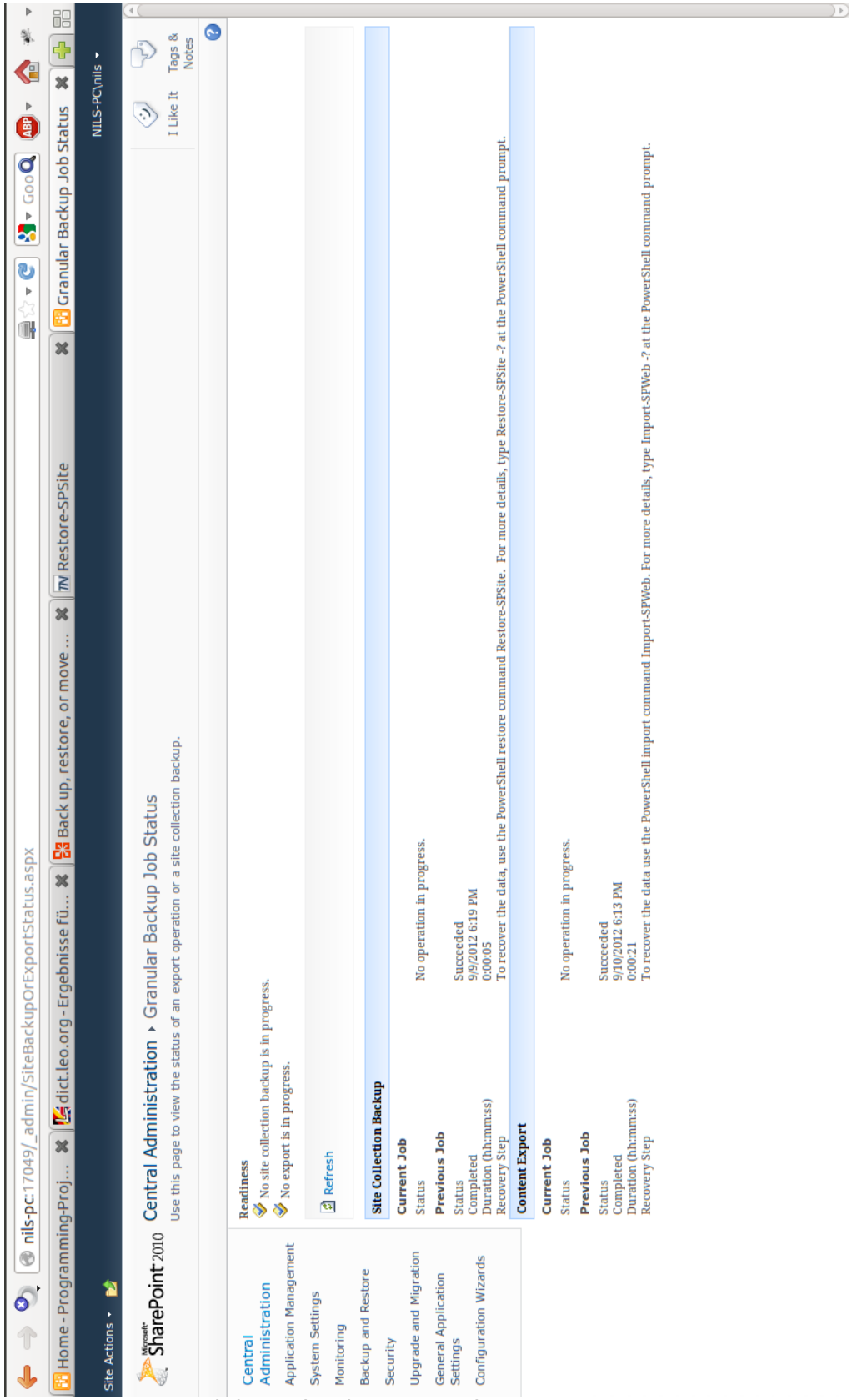
Appendix B-20: SharePoint administration interface (own source)



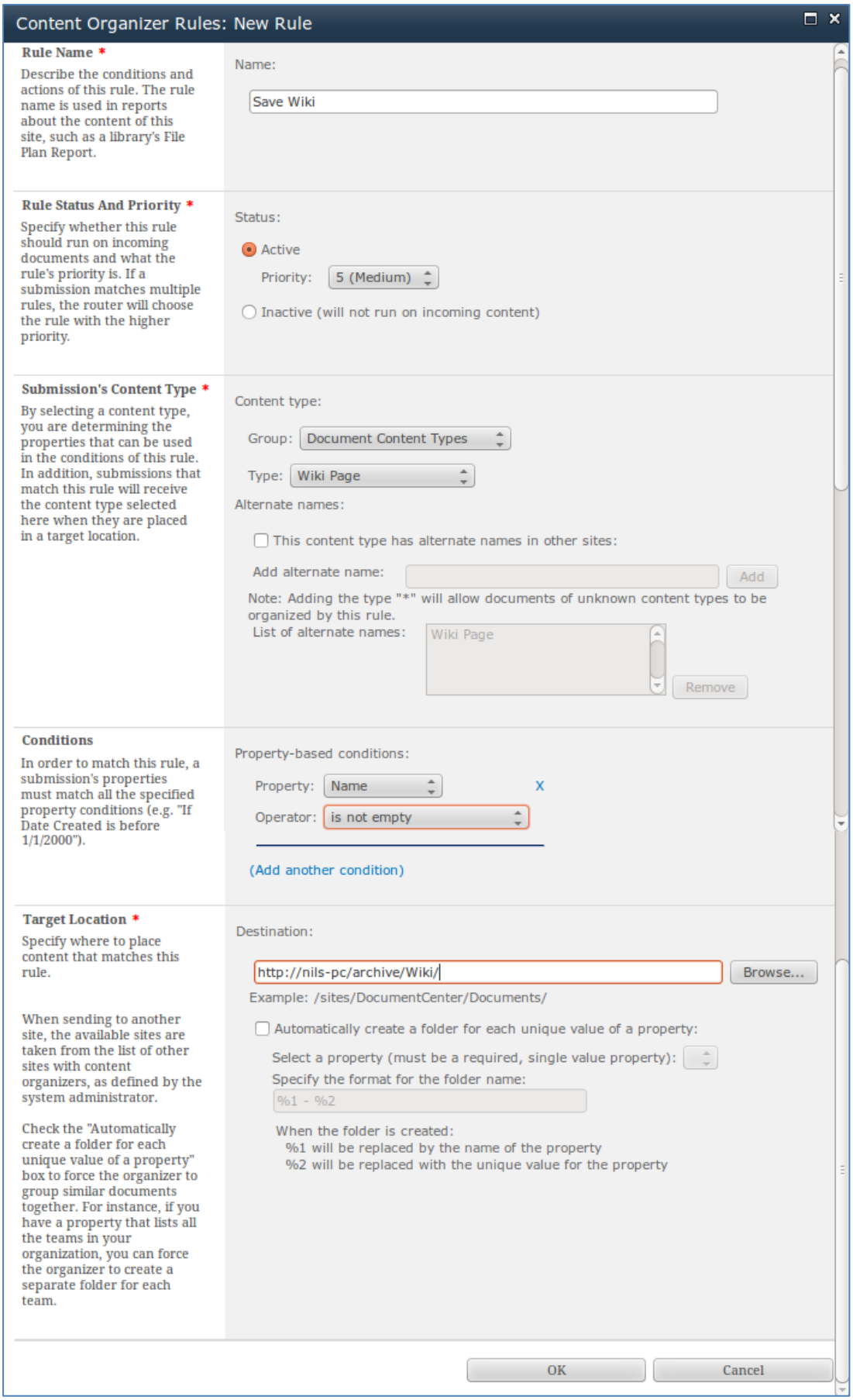
Appendix B-21: Backup and Restore functions (own source)



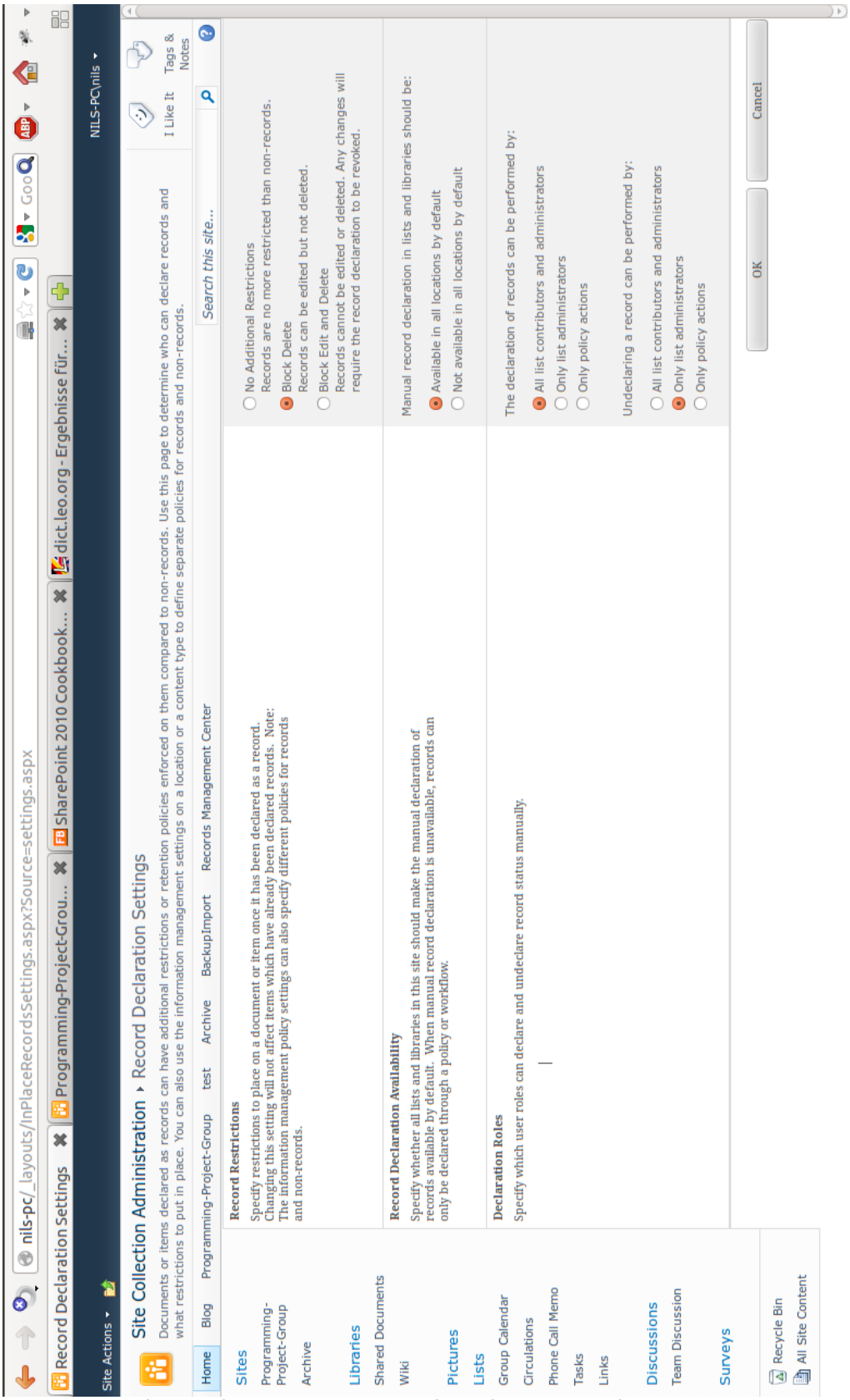
Appendix B-22: Site export options (own source)



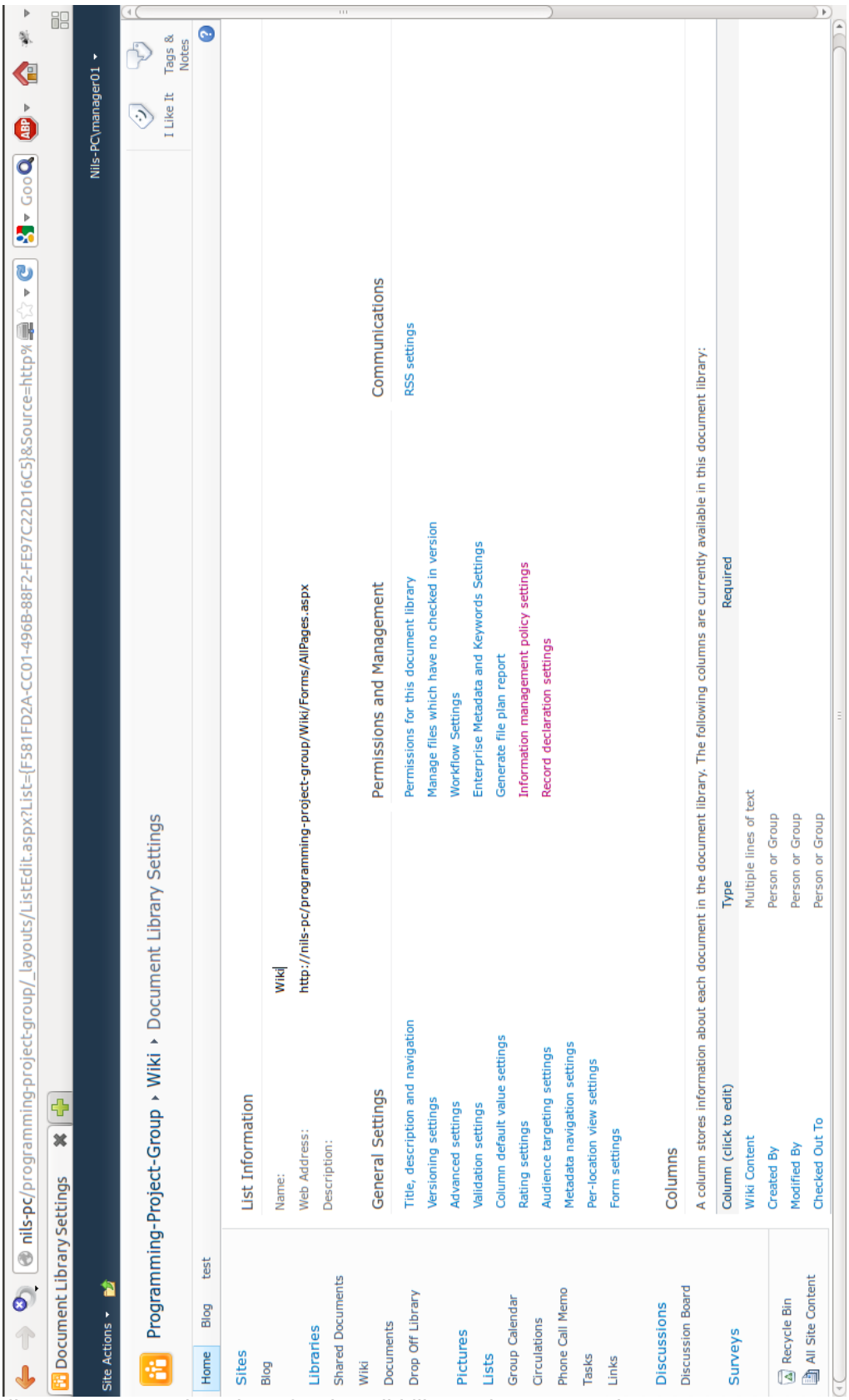
Appendix B-23: Export status information (own source)



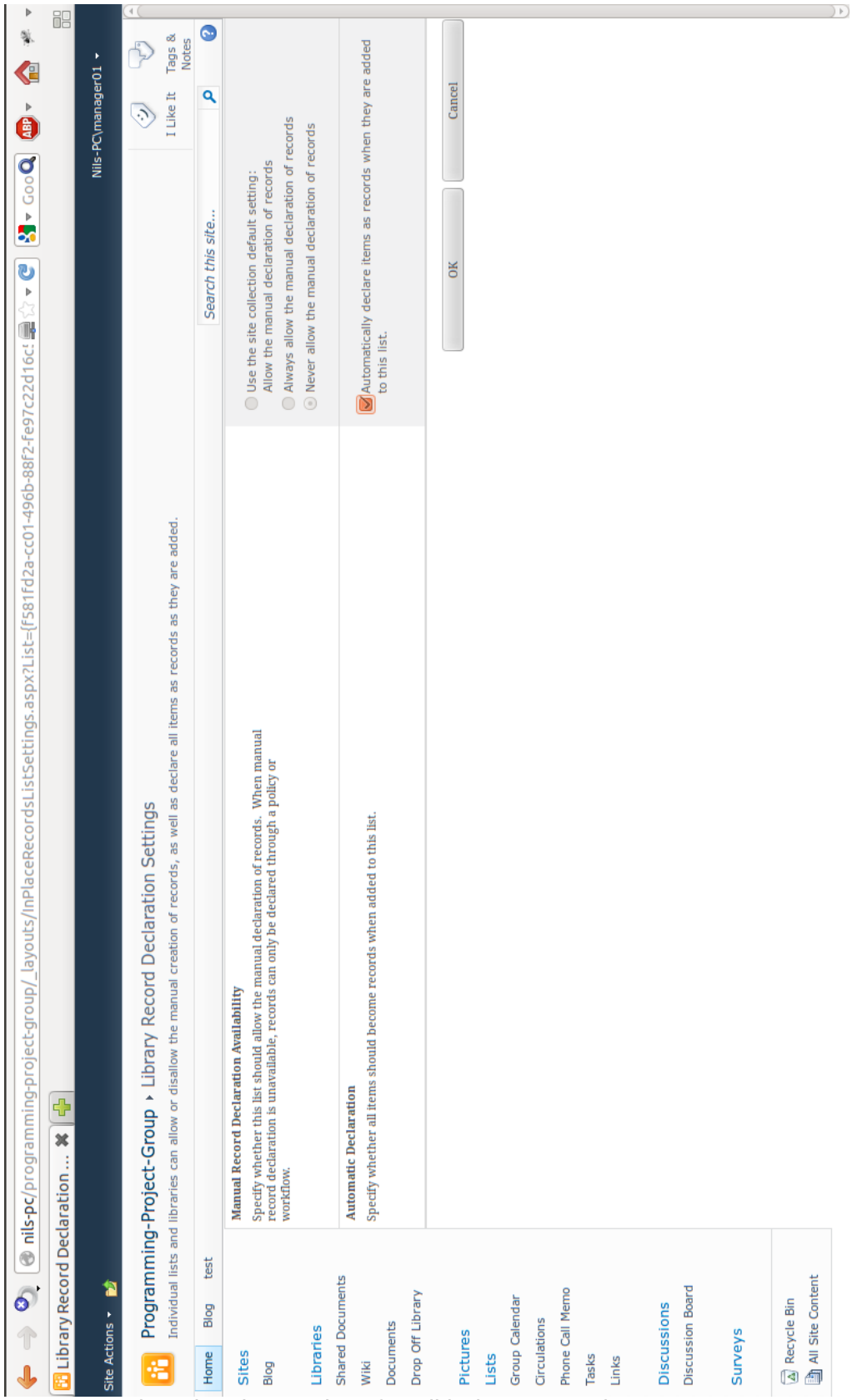
Appendix B-24: Exemplary wiki export rule (own source)



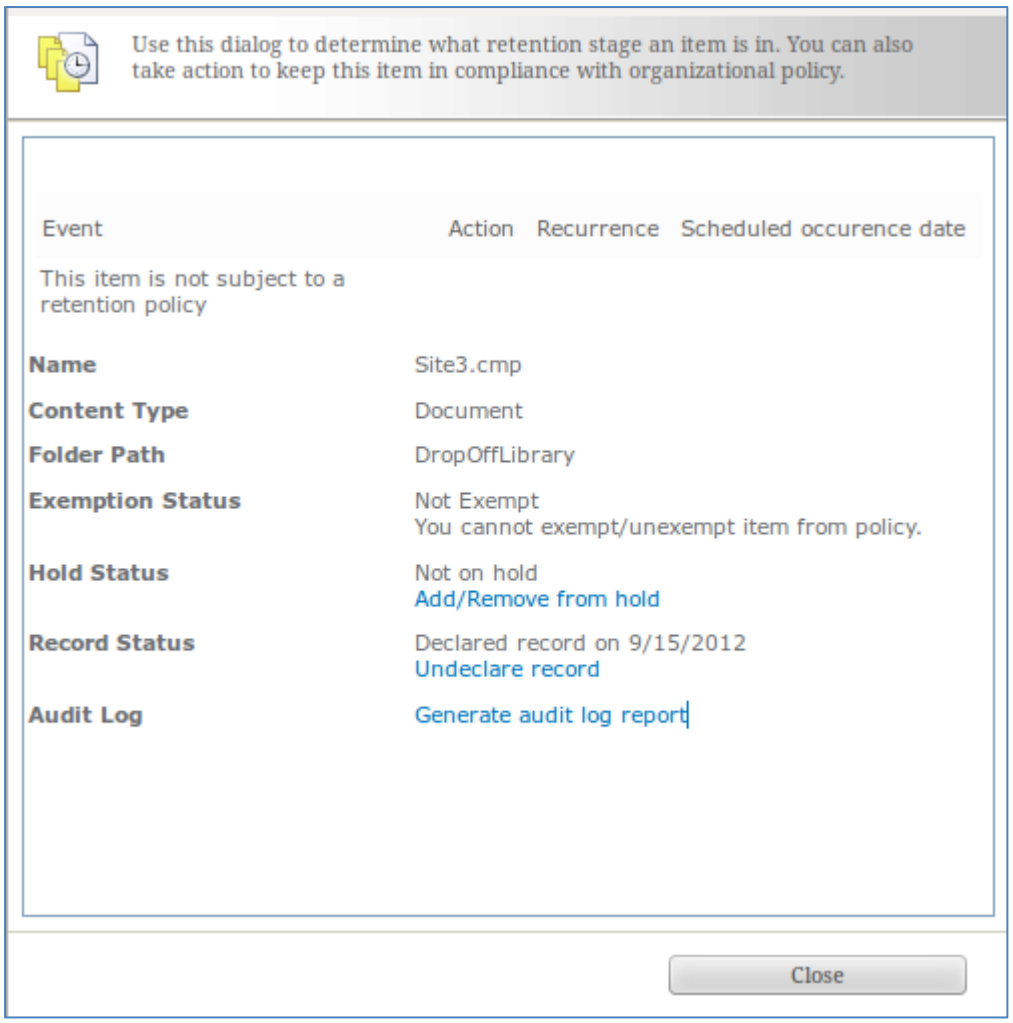
Appendix B-25: General records management settings (own source)



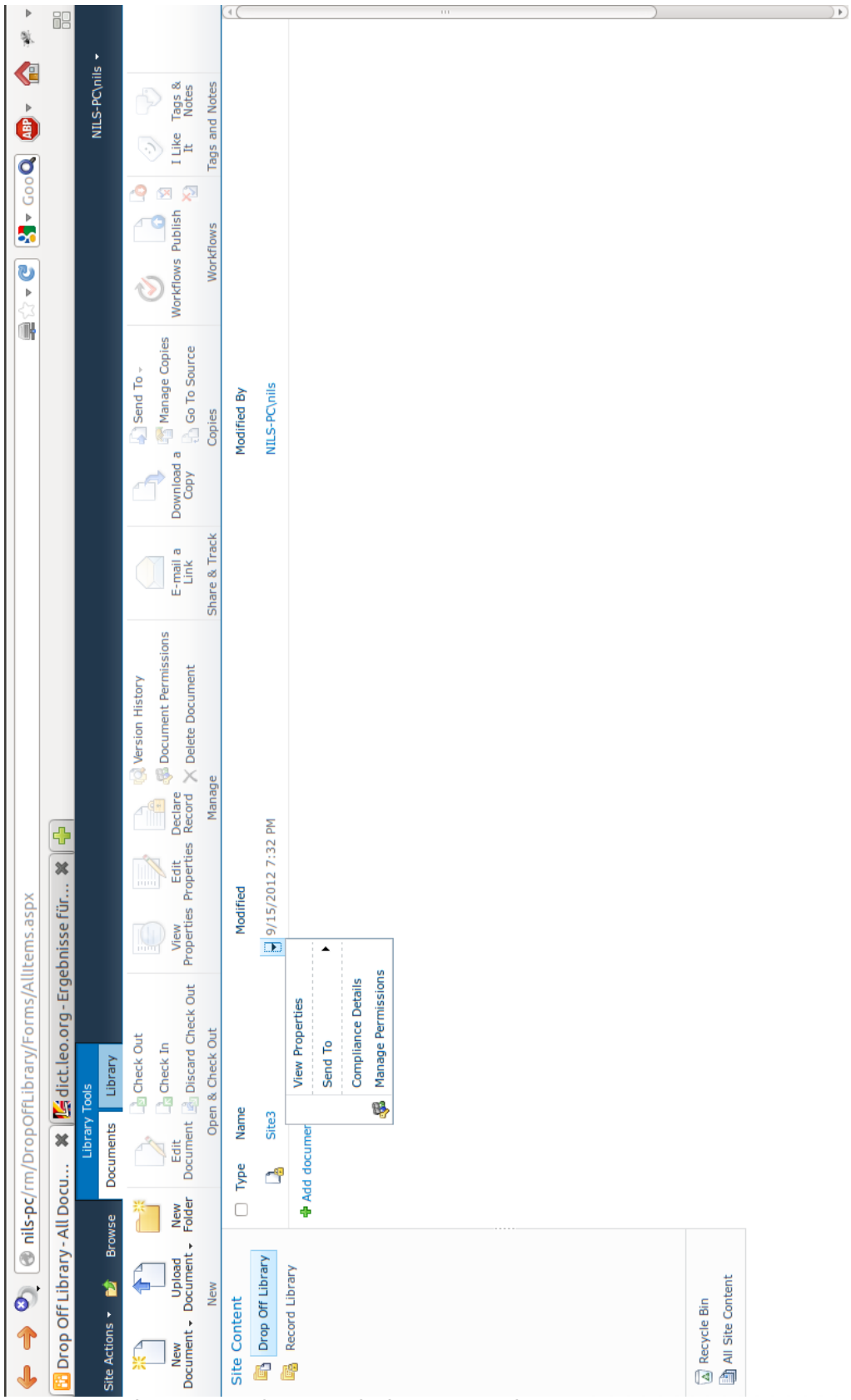
Appendix B-26: General settings for the wiki library (own source)



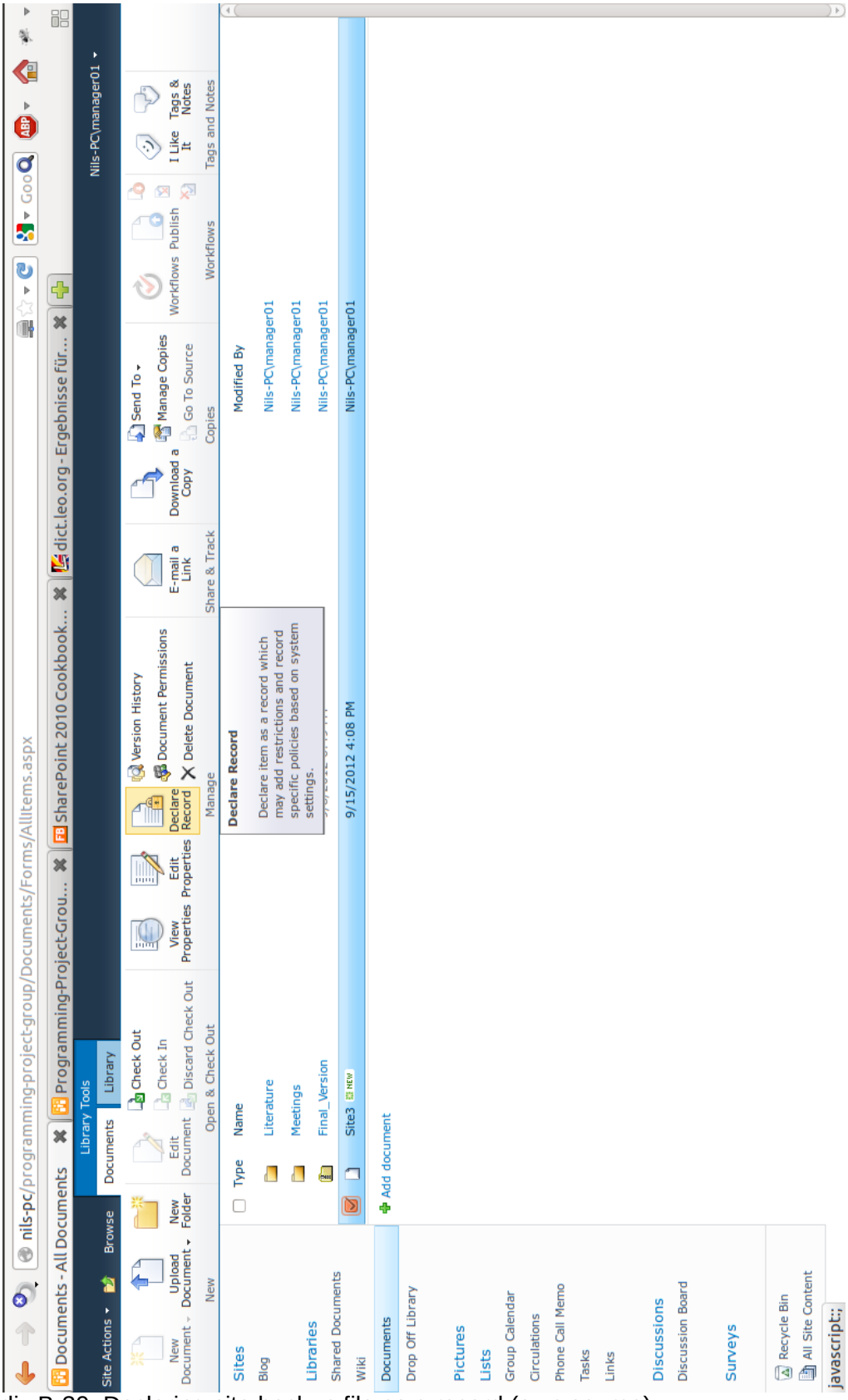
Appendix B-27: Record Declaration Settings for wikis (own source)



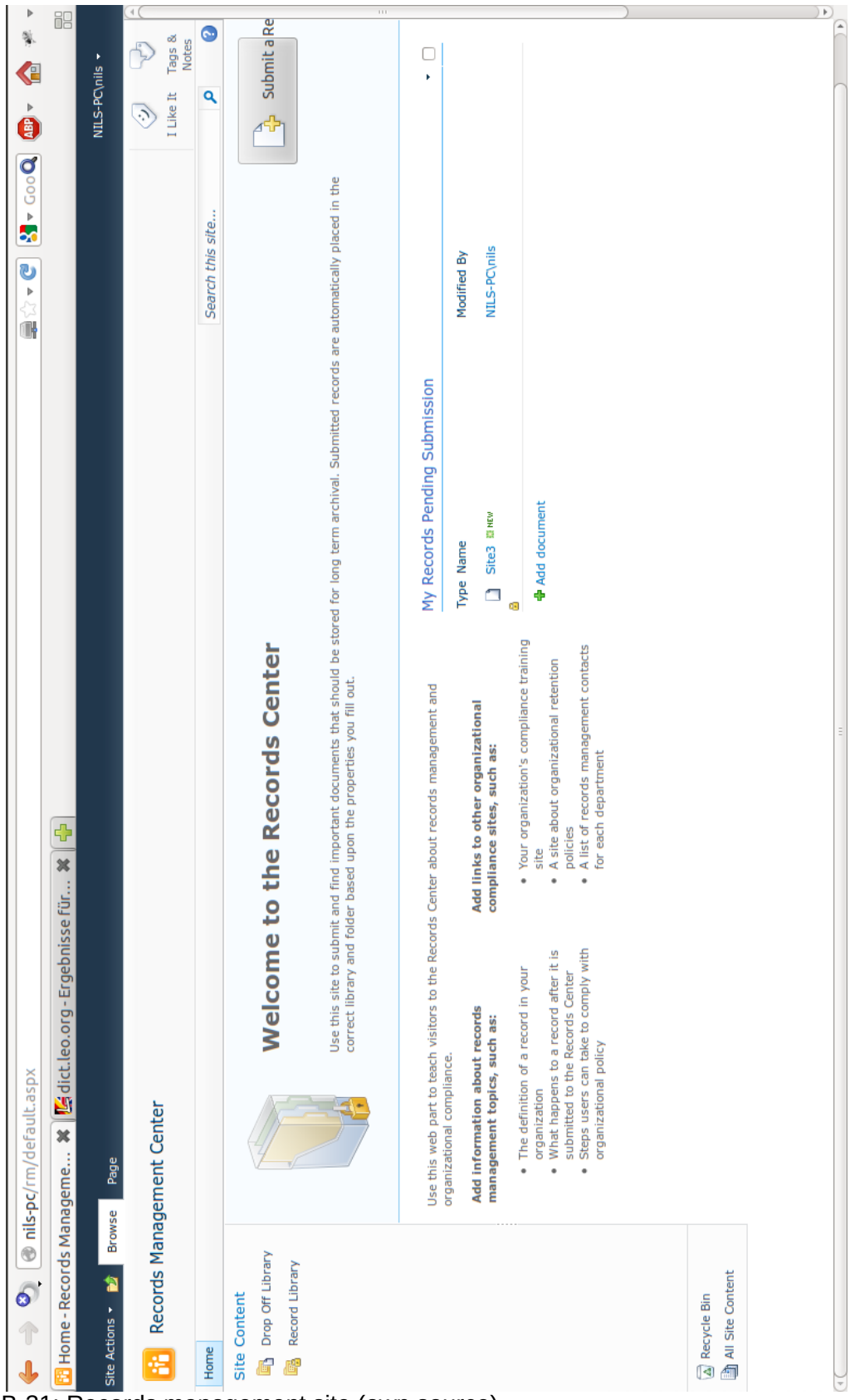
Appendix B-28: Compliance details page (own source)



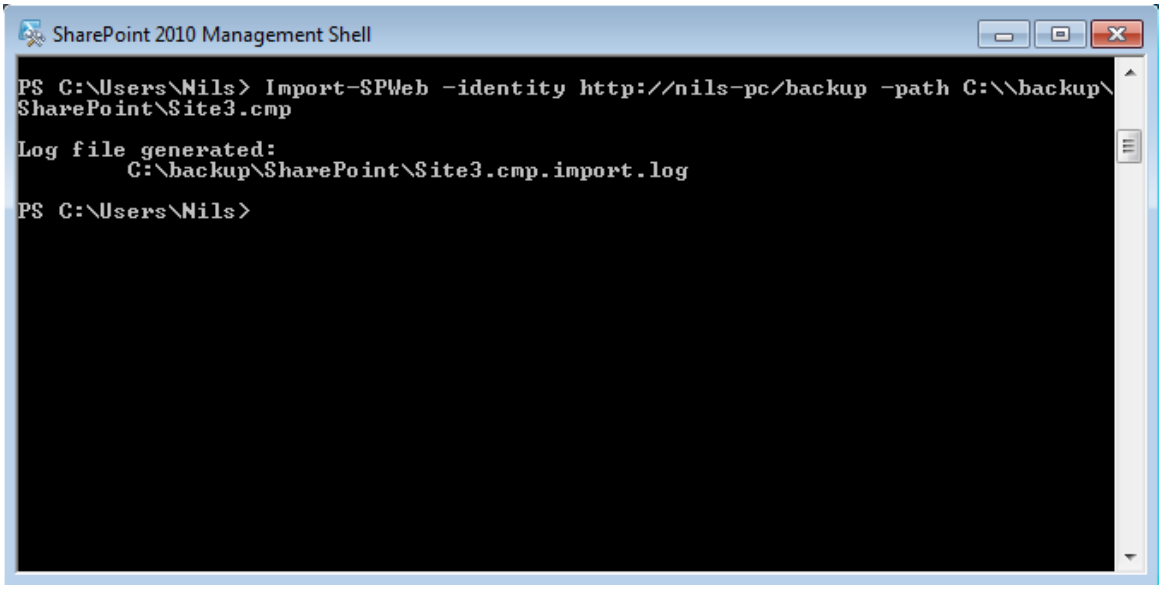
Appendix B-29: Drop-down menu for records (own source)



Appendix B-30: Declaring site backup file as a record (own source)



Appendix B-31: Records management site (own source)




```
[9/11/2012 4:48:11 PM] Progress: Import completed.
[9/11/2012 4:48:11 PM] Finish Time: 9/11/2012 4:48:11 PM.
[9/11/2012 4:48:11 PM] Duration: 00:00:46
[9/11/2012 4:48:11 PM] Total Objects: 498
[9/11/2012 4:48:11 PM] Finished with 6 warnings.
[9/11/2012 4:48:11 PM] Finished with 0 errors.
```

Appendix B-32: Site import command with log file output (own source)