

DEBORA¹ Telematics Applications Programme (project no. 5608)

DELIVERABLE 8.1

Revision: 1.4

DEBORA Client Functional Requirements

Written by:

Duncan Pemberton & David Nichols
({pemberto, dmn}@comp.lancs.ac.uk)
*Cooperative Systems Engineering Group,
Computing Department, Lancaster University*

&

Majid Ihadjadene
(ihadjade@enssib.fr)
enssib (école nationale supérieure des sciences de l'information et des bibliothèques)
Villeurbanne Cedex France

Contractual Date of Delivery: 31/5/99

Actual Date of Delivery: 28/5/99

Deliverable: D8.1

Nature of the Deliverable: RE/SP**

Deliverable Type: PU*

Dissemination Level: LI*

Work Package: WP8

Task: Client functionality specification

Responsibility: Lancaster University

*Type: PU-public, LI-limited, RP-restricted

**Nature: PR-Prototype, RE-Report, SP-Specification, TO-Tool, OT-Other

¹ Local project website, <http://www.comp.lancs.ac.uk/computing/research/cseg/projects/debora/>

Contents

1	<i>Introduction</i>	3
2	<i>Client functional requirements table</i>	3
3	<i>Requirements comparison table</i>	4
3.1	System descriptions	6
3.1.1	The Magna Carta Viewer	6
3.1.2	The Antique Book Digital Library System	6
3.1.3	Ariadne.....	7
3.1.4	The Digital Scriptorium.....	7
3.1.5	The William Blake Archive.....	7
3.1.6	HELIOS.....	7
3.1.7	Typical Library OPAC	7
3.1.8	PABN (France)	8
3.1.9	Octavo	8
4	<i>Selected application scenarios for the DEBORA client</i>	8
4.1	Matchmaking	8
4.2	Customised access & international user support	9
4.3	Annotations & document cross-references	9
4.4	Document ratings	9
4.5	Meta searching	9
4.6	Image controls	10
4.7	Privacy	10
4.8	User interaction with librarians	10
5	<i>Conclusion</i>	10
5.1	DEBORA client prototype functionality summary	11
5.2	Relationship of the client to other Workpackages	13
5.3	Implementation issues	13
6	<i>References</i>	14

DEBORA Client Functional Requirements

1 Introduction

This document presents an outline of the functionality required in the DEBORA client application cross-referenced against seven other digital library systems. To further illustrate the functionality a number of user-driven scenarios are presented for key requirement areas.

Note that in this specification we primarily describe the more “advanced” functionality of the client. The client will also operate at a basic level allowing users to search and retrieve digitised pages from books and display these in their scanned, electronic form. Therefore the reader can consider the DEBORA client to contain the majority of the applicable functionality outlined in the systems described in section 3.1, whilst expanding upon this basic feature set with the richer client functionality laid out in the following sections.

In reading this document it should be noted that the source materials are expected to be available in image form rather than in full-text format. Consequently, features such as linguistic analysis, concordances, etc. are not relevant. Such functionality may be desirable when generating the full text versions of the source material, although these features are beyond the scope of the project.

2 Client functional requirements table

The requirements are prioritised according to the following three-point scale –

- A. Essential primary application functionality
- B. Secondary application functionality
- C. “Nice-to-have” add-on functionality, i.e. features not directly affecting the application, though possibly improving certain usability aspects.

Functionality of the client may be split into the following key areas of –

- Document access
- Protocols & standardisation
- Document annotation
- Document ratings
- Information cross-referencing
- Collaboration & awareness mechanisms
- User activity history recording
- Searching
- Document image control
- Navigation & browsing control
- Privacy

Req. Type	Req. No.	Detailed Software Functional Requirement	Priority Level
Document access	1.	Distributed Internet access of data & cataloguing information	A
	2.	Auditing facility to enable librarian to track document usage and target future digitisation efforts to popular subject areas	B
	3.	Multi-mode GUI to allow customised access by end-users, library staff, and system administrators (i.e. separate audit, administration and management interfaces)	B
	4.	Scaleable solution	A
	5.	Transparent operation with the WWW, i.e. zero installation software deployment	A
	6.	Internationalisation of the user interface	A

Req. Type	Req. No.	Detailed Software Functional Requirement	Priority Level
Protocols & standardisation	7.	Use of standard image (e.g. JPEG) and data access protocols (e.g. JDBC) to target a wide user base, and allow client interoperability (e.g. Z39.50)	A
	8.	Provide document access control mechanisms	A
	9.	Dynamic server configuration. Set-up the client to locate the nearest or fastest document server depending upon network traffic conditions, thereby ensuring a good user response time.	B
Document annotation	10.	Facilities for per article & per book annotations	B
	11.	User “on-page” annotations (e.g. freehand sketch)	B
	12.	Annotation sharing & personal (i.e. private) annotation functionality	A
	13.	Highlighting (i.e. highlighter pen tool) and tagging of information on digitised pages	A
	14.	Bookmark functionality	A
Document ratings	15.	Personal bibliography creation & sharing	A
	16.	User article ratings with selectable criteria [1]	C
Information cross-referencing	17.	Related article references & user cross-referencing insertion	B
	18.	Page Markers – cross-referencing	B
Collaboration & awareness	19.	User awareness mechanisms (through observation of others, and other interactive communication facilities, e.g. subject chat forums)	B
	20.	Interaction & collaboration with librarians (via synchronous/asynchronous mechanisms)	B
	21.	Creation and dissemination of recommended reading lists when user registers an interest in a subject area	C
	22.	“Matchmaking facility” [2], allowing a searchable history mechanism of previous user activity to enable interaction between users with similar interests	B
User activity history recording	23.	Reading history storage and recall enabling the user to back track through previous sessions with full persistence of each activity	B
	24.	“Meta-Searches “, i.e. searching through previous searches	B
Searching	25.	Search automation (via intelligent wizards), persistence on previous searches (See R.24).	A
	26.	Search according to user ratings (See R.16).	C
	27.	Page/text search facility with pattern matching algorithms	B
	28.	Help facility to guide users through the initial search process	B
Image control	29.	Image rendering controls and optimum setting recall (e.g. brightness/contrast control, anti-alias retainable user preferences, etc.)	B
	30.	Image extraction from scanned document pages to allow for the use of illustrations contained in the DEBORA corpus in other works.	B
	31.	Level of detail selection to improve image download speed	B
	32.	Raise/lower image-quality control	B
	33.	Zoom-control (i.e. zoom a selectable area with full control over image magnification)	A
Navigation & browsing control	34.	Selectable thumbnail index of pages	B
	35.	Active indexes (i.e. clicking on a digitised entry moves to indexed page)	B
	36.	Intuitive page/book navigation display controls	A
	37.	Page goto control	A
Privacy	38.	User interface for administrators to inspect and create reports on user activity within the system (See R.2).	B
	39.	User inspection of stored personal information	B

Table 1 - DEBORA Client – High-level software functional requirements

3 Requirements comparison table

Table 2 - KEY TO SYMBOLS

✓	Feature Supported
✗	Feature Not Supported
☒	N/A
☑	May be provided

The above requirements are cross-referenced with those provided in five other digital library systems (See Table 2 below). A brief overview of each system is provided in the subsections that follow the table.

DEBORA Req. No.	Magna Carta Digital Viewer ¹	Antique Book Digital Library ²	Ariadne ³	Digital Scriptorium ⁴	The William Blake Archive ⁵	HELIOS ⁶	Typical OPAC ⁷	BNF ⁸ (PABN)	OCTAVO ⁹
1.	✓	✓	✓	✓	✓	✓	☑	✓	✗
2.	☒	☒	✓	☒	✗	✓	✓	✓	✗
3.	☒	☒	✗	✗	✗	✓	✓	✓	✓
4.	✓	✓	✓	✓	✓	✓	✓	?	?
5.	✓	✓	✓	✓	✓	✓	☑	✓	✗
6.	✗	✗	✗	✗	✗	✗	✗	✗	✓
7.	✓	✓	✓	✓	✓	✓	✓ (Z39.50)	✗	✗
8.	✗	✗	☒	✗	✗	✗	✗	✗	✓
9.	✗	✗	☒	✗	✗	✗	✗	☒	✗
10.	✗	✓ (per book)	☒	✗	✓	✗	☑	✓	✗
11.	✗	✗	☒	✗	✓	✗	✗	✓	✗
12.	✗	✗	✓	✗	✓	✗	✗	✗	✗
13.	✗	✗	☒	✗	✗	✗	✗	✓	✓
14.	☒	✗	☒	✗	✗	✗	✗	✓	✓
15.	☒	✗	✓	✗	✗	✗	☑	✗	☒
16.	☒	✗	☒	✗	✗	✗	✗	✗	☒
17.	✗	✗	☒	✗	✗	✓	☑	✗	☒
18.	☒	✗	☒	✗	✗	✗	✗	☒	☒
19.	✗	✗	✓	✗	✗	✗	✗	✗	✗
20.	☒	✗	✓ (Asynchronous)	✗	✗	✗	☑ (E-mail)	✗	✗
21.	☒	✗	✓	✗	✗	✗	✗	✗	✗
22.	☒	✗	✓	✗	✗	✗	✗	✗	✗
23.	☒	✗	✓	✗	✗	✗	✗	✓	✗
24.	☒	✗	✓	✗	✗	✗	✗	✓	✗
25.	☒	✗	✓	✗	✓	✗	✗	✓	✗
26.	☒	✗	✗	✗	✗	✗	✗	✗	✗
27.	✗	✓	☒	✓	✗	✓	✗	✗	✗
28.	☒	✗	✗	✗	✗	✗	✗	✓	✓
29.	✗	✗	☒	✗	✗	✗	☒	✓	✓

¹ See, http://www.bl.uk/diglib/diglib_main.html

² See, <http://www.antiquebooks.net/readpage.html>

³ See, <http://www.comp.lancs.ac.uk/computing/research/cseg/projects/ariadne/>

⁴ See, <http://sunsite.berkeley.edu/scriptorium/form.html>

⁵ See, <http://www.iath.virginia.edu/blake/>

⁶ See, <http://heinzl.library.cmu.edu/HELIOS/>

⁷ OPAC - On-line Public Access Catalogue

⁸ Note that Gallica does not support annotation features. (see, <http://gallica.bnf.fr>)

⁹ See, <http://www.octavo.com>

DEBORA Req. No.	Magna Carta Digital Viewer ¹	Antique Book Digital Library ²	Ariadne ³	Digital Scriptorium ⁴	The William Blake Archive ⁵	HELIOS ⁶	Typical OPAC ⁷	BNF ⁸ (PABN)	OCTAVO ⁹
30.	x	x	☒	x	x	x	☒	x	x
31.	x	x	☒	✓	x	✓	☒	✓	✓
32.	x	✓	☒	x	x	x	☒	x	✓
33.	✓	x	☒	x	✓	x	☒	✓	✓
34.	☒	x	☒	✓	✓	x	☒	x	✓
35.	☒	✓	☒	x	x	x	☒	x	✓
36.	✓	✓	☒	x	✓	✓	☒	✓	✓
37.	☒	✓	☒	x	x	✓	☒	✓	✓
38.	☒	x	✓	☒	x	✓	✓	x	x
39.	☒	x	☒	☒	x	x	x	x	x

Table 2 – DEBORA requirements cross-referenced with other digital library systems

3.1 System descriptions

This section provides a brief description of the seven systems outlined in Table 2 above.

3.1.1 The Magna Carta Viewer

The Magna Carta viewer displays a scanned digital copy of the Magna Carta document on the WWW. The system features selectable areas of the document image, that when clicked upon display a zoomed section of the main document. As the document presented only consists of a single page much of the functionality that applies to DEBORA does not apply to this particular system. For example, bookmarking functionality, etc. is not required for single page document based systems.

Once zoomed in, the Magna Carta document may be further navigated via four arrows moving the viewed image to the next section of the page. Finally, the user may zoom out to the full-page view with another control. The system therefore presents a simple, functional and usable viewing mechanism to allow widespread availability of the Magna Carta document across the Web.

3.1.2 The Antique Book Digital Library System

Provides a more sophisticated set of functionality than the Magna Carta viewer again using the WWW [3]. A selected number of books are available scanned and presented to the user in electronic form. The system allows sharable user annotations for each digital book, and the user the functionality to add their own comments.

Basic persistence is implemented allowing the ability to return to previous reading locations if the digital library is left and the user chooses to return at a later date to complete reading. This system like the Magna Carta viewer presents a simple navigation interface, also including a skip to page control.

An active index is provided where clicking on the table of contents control presents the original book's scanned index pages. Another click on the chapter will display the actual page containing the scanned images of the selected chapter.

To aid page download speed an image quality control is included to either reduce or enhance image quality. Finally, OCR of the scanned pages has been used to create a keyword searching facility of the text contained on the document pages. The pages where the search text is located are displayed and the user may navigate directly to each page via a mouse click.

3.1.3 Ariadne

Ariadne [4] provides support for the visualisation of search sessions, with the aim of making it easier for users to reflect, share and comment on their understanding with others.

Ariadne enables searches of digital libraries to be,

- Visualised
- Communicated
- Edited
- Annotated

Digital library systems should provide support for collaboration, including awareness of other users and consideration of the physical attributes of libraries that may be lost when using a computerised alternative. Ariadne therefore tries to address these issues through search analysis software.

Ariadne includes records of all previous user searches. This information can be used to locate the most valuable resource of all - people with similar interests by browsing through provided lists. Many searches duplicate previous searches - yet the information is normally lost and new searchers have to "re-invent the wheel." Ariadne supports the re-use of searches to take advantage of other searcher's efforts to retrieve additional records. This is known as *social information filtering*. Ariadne can also be used as a teaching tool to go over with a student what has been done, and other things that might be done to improve searching productivity.

Additional features of the Ariadne tool support includes, the annotation area that allows the addition of user annotations to searches, and the Email facility that provides the capability for an email message to be sent to another user regarding the current search.

3.1.4 The Digital Scriptorium

The Digital Scriptorium delivers document pages depending upon searches of various data types including: country, binding, watermark, title, author, and shelf-mark, etc. Scanned images of the documents are then returned in a WWW page with a selectable image size. This system does not feature any book-marking or CSCW [5] functionality. The digital scriptorium system is a search and retrieval only type system.

3.1.5 The William Blake Archive

The William Blake Archive [6] is notable in its inclusion of a Java based image viewer applet in the digital library system. The applet allows users to zoom in to a precisely scaled version of the original image and add or edit annotations that are connected to a specific region of the image file. The use of Java improves the usability of the system and makes the inclusion of such interactive features possible.

3.1.6 HELIOS

HELIOS [7] is notable in that it provides limited guidance through the document search process, via an HTML based hierarchical browsing structure. The database may also be browsed according to search terms. Once the document is delivered in the web browser controls are available to, display the document's location in the database hierarchy, move between documents & pages, goto pages, and enlarge & shrink images.

3.1.7 Typical Library OPAC

OPAC systems are generally used to search large databases of library cataloguing information. Typically advanced search mechanisms are provided including author searches, keywords, etc. Additionally an OPAC may allow lists to be produced of books available by subject area.

Other facilities that are often implemented include the creation of personal bibliography facilities; these may also be shared with other users in certain OPAC systems.

OPAC systems may have WWW clients or telnet based operation. The Z39.50 protocol is often used in their implementation. An OPAC is therefore simply a “cataloguing and search” software system. OPACs generally do not provide CSCW facilities and electronic document delivery mechanisms that are required by DEBORA.

3.1.8 PABN (France)

PABN contains two kinds of interfaces, simple and normal. The major difference between these two access methods is the possibility of reading a number of different documents at one time (e.g. using a MDI¹ interface). A user can also move the windows, helping to create an optimal reading environment.

PABN also contains the following annotation facilities. Users can annotate text either,

1. Directly on the page
2. Using bloc-notes, as in Microsoft Word

Additionally PABN supports,

1. Bookmarking
2. Highlighting and tagging of page information
3. Modifying or deleting annotations
4. Emailing & printing annotations
5. Searching of the annotations
6. Ranking of the annotations

Each annotation itself consists of a,

1. Title
2. Icon and/or a colour
3. The commentary text

3.1.9 Octavo

Most of the OCTAVO features exist in other DL systems. Two features that are unique are, the possibility of reading the translations of some books, and the use of two specialised indexes. The *marginalia index*: for abbreviations, and the anecdotes index.

4 Selected application scenarios for the DEBORA client

The following subsections present a set of user driven scenarios outlining the rationale and envisaged operation behind selected representative DEBORA requirements. Requirements described by the scenarios are cross-referenced with Table 2.

4.1 Matchmaking

Cross-references requirement(s): R.22.

A typical digital library (DL) user, to save effort and pool resources, wants to find a person with similar research subject interests. Consider the scenario presenting a rationale for computer based support for this functionality,

You work in a small research institution/university with a well-stocked library. You are browsing the shelves when you spot a colleague from a different department looking at books in ‘your’ section of the library. You strike up a conversation and discover that this colleague is doing some research in a new area for her. She discovered a reference to a book and was in the process of tracking it down. You discover that you have a mutual interest in the topic, although you are viewing from the perspective of two quite different disciplines. Amusingly

¹ MDI – Multiple document interface

you discover that in these two disciplines the topic has two quite different names, so you would not have known about your mutual interests from reading the organisation's staff information guide. You decide to share resources and tackle the problem together.

4.2 Customised access & international user support

Cross-references requirement(s): R.2, R.3, R.6, and R.37.

For the DL system to be usable in terms of response time and language presentation the client software needs to select the fastest server and appropriate language interface to present to the user. To illustrate this consider the following scenario,

A user in a distant country requires access to the digital library. They do not speak very much English and their Internet connection to the UK is intolerably slow. The client software on the user's behalf selects the appropriate local language to display the user interface in. A check is then performed for a mirror of the database nearer to their local site. The native language interface is then presented with the fastest Internet connection found.

4.3 Annotations & document cross-references

Cross-references requirement(s): R.10, R.11, R.12, and R.13.

Users need to add information to electronic digital library books as they would do by making notes from a book loaned from a traditional library. Consider the following scenario,

A user has found a particularly interesting section of a book that they wish to initially highlight before carrying on reading. After completing their reading of a particular chapter they then come back to the highlighted section. The user makes additional notes using drawing tools supplied with the DL client software directly on the page. In case any other users of the digital library are interested in this annotation the user then chooses to share it. Another similarly interested reader then searches the DL and notices this annotation. They are then led to some other connected literature that previously they were unaware of. The initial reader placed this cross-reference in their annotation. Clicking on the reference displays the related article.

4.4 Document ratings

Cross-references requirement(s): R.16.

Ratings may be obtained for documents in a digital library in one of two ways. Firstly, the user may assign a rating to a document explicitly, or the rating may be implicit, created by the system assigning automated criteria to construct the rating. For example, taking into account the number of accesses, amount of time read, etc. To illustrate this consider the following scenario,

A user requires a document giving background detail to their research area. To do this they consult a digital library system. The user wants the best regarded work to start their literature review, they therefore use the digital libraries' rating system to help them select where to start reading. The system assigns an implicit rating to items that have not been explicitly rated. The user may choose to override the implicit ratings if someone else has previously personally rated the document.

4.5 Meta searching

Cross-references requirement(s): R.23, R.24.

Users often need breaks between reading information to give them a chance to further reflect upon it. It is therefore important to record user activity in case a user needs to backtrack through their previous work at some later date. Consider,

After spending many hours searching through the digital library the user leaves the library client. Some weeks later they return to continue their research. Being a regular visitor to the library they have generated a large search history. Therefore to quickly locate previous documents and reading positions they search through their previous usage history to guide them back to their ongoing threads of research.

4.6 Image controls

Cross-references requirement(s): R.29, R.30, R.31, R.32.

Many renaissance documents are very detailed. This requires specialised functionality to enable them to be effectively studied in electronic form. Consider the scenario,

After locating a particularly interesting page in a renaissance document a user wishes to study a specific section of the page in more detail to ascertain the fine details of an illustration. To do this they select an area of the page for enlargement and a scale factor to magnify the image by. To try and further improve the image quality the user then selects a setting for improved level of detail (i.e. post-processing image enhancement). This setting is available in conjunction with another default setting optimised for download speed. The user may customise these settings possibly adding their own to improve the image quality on a particular page by changing the anti-alias, contrast, colour temperature, gamma, and brightness settings of the document viewer.

4.7 Privacy

Cross-references requirement(s): R.37, R.38.

To protect the user's privacy in a cooperative digital library system the following user scenario may arise. Consider,

A legitimate fear that users may have of a matchmaking system (see section 4.1) is that although they assent to the loss of privacy necessary to enable it to work, they fear that the data provided may in the future be put to other, unauthorised uses. Consequently they may refuse to use a system, irrespective of the benefit which they are in perfect agreement it would provide. Many of these concerns about personal information derive from a feeling of lack of control caused by the intangibility, ease-of-duplication and the "black box" nature of computer storage.

4.8 User interaction with librarians

Cross-references requirement(s): R.20.

It is important to provide mechanisms in the digital library where the user can interact with people of experience, as with the traditional library "help desk". The scenario presented below describes the rationale for the need to carry through this functionality into the digital library,

Collaboration between users and staff in traditional libraries is most evident at the help or reference desk. Users' problems are (hopefully) resolved through interaction that for the most part relies on both people being present in the same time & place. Libraries have often provided additional services to support users who are not physically in the library. Remote interaction has occurred for some time, using letters, internal mail and particularly the telephone. One of the simplest computerised approaches to the same issue has been to set up an email reference desk service. To better replicate "same time and place" interactions in digital libraries a synchronous communication forum should be provided. Ideally trained and knowledgeable staff should operate this.

5 Conclusion

This document has presented the high-level functional requirements of the DEBORA client system. A prioritised table of requirements has been supplied that is cross-referenced with functionality provided

in other similar DL (Digital Library) based systems. Sets of sample usage scenarios were also developed to help illustrate the user-based perspectives behind the requirements. The scenarios are cross-referenced with the traceable requirements elicited at the start of the document.

As can be seen from the requirement tables DEBORA presents some novel functionality to help improve user access to digital library based documents. This functionality should be prototyped to assess its user impact in each case.

5.1 DEBORA client prototype functionality summary

Table 3 (below) shows a summary of the functionality to be included in the DEBORA client prototype. It is envisaged that this will be carried out in three separate stages. Firstly the critical features of the system's operation will be included. These will then be followed by the secondary and tertiary functionality of the client in subsequent prototype releases (see the table below for details). At each stage various software "solutions" to the functionality will be assessed. The solution deemed to be most appropriate will remain in the system through the next prototype "round".

Table 3 - KEY TO SYMBOLS

- ✓ Feature targeted for immediate inclusion in the DEBORA prototype
- ☑ Feature targeted for implementation after the primary features
- ❖ Tertiary prototype feature

Prototype Inclusion Level	DEBORA Software Functionality
✓	Distributed Internet access of data & cataloguing information. Prototyped to include tight WWW integration.
✓	Scaleable solution. Initial prototype to deal with a small database of information. Scalability of the chosen architecture will be assessed at this prototype stage.
✓	Transparent operation with the WWW, i.e. zero installation software deployment. Evaluation of interface designs in Java embedded into web pages and via separate applet 'JFrames'.
✓	Internationalisation of the user interface. Addition of preference options for the default GUI language presentation.
✓	Use of standard image and data access protocols to target a wide user base, and allow client interoperability (e.g. Z39.50)
✓	Provision of document access control mechanisms
✓	Annotation sharing & personal (i.e. private) annotation functionality. Add privacy to features via personal password authentication.
✓	Highlighting (i.e. highlighter pen tool) and tagging of information on digitised pages. Prototype various implementations of this.
✓	Prototyping of bookmarking functionality
✓	Personal bibliography creation & sharing
✓	Search automation (via intelligent wizards), persistence on previous searches. Add to prototype server-side persistence mechanism for user profiles.
✓	Zoom-control (i.e. zoom a selectable area with full control over image magnification)
✓	Intuitive page/book navigation display controls. Prototyped in client GUI.
✓	Page goto control
☑	Auditing facility to enable librarian to track document usage and target future digitisation efforts to popular subject areas
☑	Multi-mode GUI to allow customised access by end-users, library staff, and system administrators (i.e. prototyping of initial audit, administration and management interfaces)
☑	Dynamic server configuration. Set-up the client to locate the nearest or fastest document server depending upon network traffic conditions, thereby ensuring a good user response time.
☑	Prototype implementations for per article & per book user annotations
☑	Prototype "on-page" annotations (i.e. freehand sketch tools)
☑	Related article references & user cross-referencing insertion – add to annotation mechanism from previous prototype round
☑	Page Markers – cross-referencing. Prototype implementation and GUI construction.
☑	User awareness mechanisms (through observation of others, and other interactive communication facilities, e.g. subject chat forums). Prototype will evaluate each mechanism.
☑	"Matchmaking facility" [2], allowing a searchable history mechanism of previous user activity to enable interaction between users with similar interests
☑	Reading history storage and recall enabling the user to back track through previous sessions with full persistence of each activity
☑	"Meta-Searches ", i.e. searching through previous searches
☑	Page/text search facility with pattern matching algorithms
☑	Help facility to guide users through the initial search process
☑	Image rendering controls and optimum setting recall (e.g. brightness/contrast control, anti-alias retainable user preferences, etc.) Add settings to server side user profile prototype.
☑	Level of detail (LOD) selection to improve image download speed
☑	Prototype of raise/lower image-quality control
☑	Prototype of selectable thumbnail index of pages
☑	Active indexes (i.e. clicking on a digitised entry moves to indexed page)
☑	User interface for administrators to inspect and create reports on user activity within the system.

Prototype Inclusion Level	DEBORA Software Functionality
☑	Image extraction from scanned document pages to allow for the use of illustrations contained in the DEBORA corpus in other works.
☑	User inspection of stored personal information (i.e. user profiles)
☑	Interaction & collaboration with librarians (via synchronous/asynchronous mechanisms)
❖	User article ratings with selectable criteria [1]
❖	Creation and dissemination of recommended reading lists when user registers an interest in a subject area – prototype mechanism.
❖	Search according to user ratings. Prototype to determine criterion selection.

Table 3 – DEBORA prototype functionality inclusion

5.2 Relationship of the client to other Workpackages

The proposed functionality of the client in this document has been derived from several sources: work done before the project, other systems we have identified in the literature and general principles of CSCW. We believe that a client embodying the functions described here would be appropriate for the DEBORA project. However the usual development timetable for a cooperative system would be to derive the specifications from investigation of the users' requirements (WP2). Although we are presenting new technology (functionality that does not directly flow from their day-to-day work practices) we would have preferred to have the results of WP2 available when writing this document. Consequently it is possible that our perceived ordering of the importance of different features in the client may need to be revised. We do not expect the specification to change significantly but some re-prioritisation may take place to better reflect the concerns of the user.

5.3 Implementation issues

Although we have considered the client functionality in 'pure' terms it is also necessary to examine the computational issues in the implementation. As we are envisaging deploying the system via the Internet, and the World Wide Web in particular, we need to examine what system specification we can expect users to have available.

The implementation of the software functionality is likely to be a "Java pure" solution. We have noted that many less interactive parts of the client may be provided in other embedded HTML technologies such as CGI/Perl, or JavaScript depending upon the specific feature. Although many features could be implemented without the use of Java, at this stage it appears that Java will be necessary to implement advanced CSCW functionality.

The envisaged minimum machine specification necessary to run the Java client software for this project will be assumed to consist of at least a,

486/DX or better (PII200 recommended), Windows95/98/NT4.0, 32Mb RAM (48Mb recommended).

Development will be using the Java2 platform specification. By deployment we foresee that this platform level should not be a great problem, and in fact a computer capable of running a modern web browser such as Microsoft IE4/5 or Netscape's Navigator should already meet these requirements.

6 References

- [1] David M. Nichols: "Implicit Rating and Filtering" *Proceedings of the Fifth DELOS Workshop on Filtering and Collaborative Filtering*, Budapest, Hungary, 10th-12th November 1997, ERCIM, 31-36, ISBN: 2-912335-04-3.
<http://www.comp.lancs.ac.uk/computing/research/cseg/projects/ariadne/docs/delos5.html>
- [2] David M. Nichols & Michael B. Twidale: "Matchmaking and Privacy in the Digital Library: striking the right balance" *Proceedings of the 4th UK/International Conference on Electronic Library and Visual Information Research (ELVIRA 4)*, Milton Keynes, 6th-8th May 1997, Aslib: London, UK, 31-38, ISBN 0851424015.
<http://www.comp.lancs.ac.uk/computing/research/cseg/projects/ariadne/docs/elvira97.html>
- [3] Robert Thibadeau & Evan Benoit: "Antique Books" *D-Lib Magazine*, September 1997, ISSN 1082-9873.
<http://www.dlib.org/dlib/september97/thibadeau/09thibadeau.html>
- [4] Michael B. Twidale, David M. Nichols & Chris D. Paice: "Browsing is a Collaborative Process" *Information Processing & Management*, **33** (6), 1997, pp. 761-83.
<http://www.comp.lancs.ac.uk/computing/research/cseg/projects/ariadne/docs/bcp.html>
- [5] Michael B. Twidale & David M. Nichols: "A Survey of Applications of CSCW for Digital Libraries", CSEG Technical Report, Lancaster University (CSEG/4/98), 1998.
<http://www.comp.lancs.ac.uk/computing/research/cseg/projects/ariadne/docs/survey.html>
- [6] Morris Eaves: "Behind the Scenes at the Blake Archive" DRH97, Conference on digital resources in the humanities, St. Anne's College, Oxford University, Oxford, England, 14th-17th September 1997.
<http://www.press.umich.edu/jep/03-02/blake.html>
- [7] Galloway, Edward A. & Gabrielle V. Michalek: "The Heinz Electronic Library Interactive Online System (HELIOS): Building a Digital Archive Using Imaging, OCR, and Natural Language Processing Technologies" *The Public-Access Computer Systems Review* 6, no. 4 (1995).
<http://info.lib.uh.edu/pr/v6/n4/gall16n4.html>