

## ▼ Edukalibre groupware tools

**Edukalibre: Libre Software Methods for E-Education**

**110330-CP-1-2003-1-ES-MINERVA-M**

Prepared by the University of Leeds in collaboration with all partners.

**Authors: Vania Dimitrova, Chris Tebb, University of Leeds.**

### ▼ 1 Introduction

The Leeds University contribution to the Edukalibre code will be a set of web based groupware tools that integrate with the [moodle](#) learning environment. This document defines the functional requirements of these groupware tools, then goes on to define a functional specification and project timescale.

### ▼ 2 Functional requirements

The initial proposal for the Edukalibre project states that the project should provide:

"A set of tools providing basic e-learning capabilities, built around the tools for collaborative editing. Those tools should mimic those of systems like WebCT or BlackBoard, but supporting a broader interaction among groups of professors and students not related to the same institution."

One of Leeds' contributions to this project is "to collaborate with other groups in the development of new products needed by the project". At the kick-off meeting in Madrid, we agreed that this contribution will concern the development of appropriate tools for group work needed by Edukalibre partners and the higher education learning community, in general.

The report on pedagogical studies to gather design requirements also stressed the importance of usable groupware tools. Recent comments sent by the group in Karlsruhe highlight the importance of having general groupware tools that are "intuitive" to use both by instructors and learners. It is made clear that groupwork will be the key part of their case study. In the same vein, the team from Lugano stressed the importance of groupware for their case study and commented on the poor usability of the groupware tools in Moodle.

The questionnaire that gathered the requirements of the Edukalibre partners suggests that group discussions, document access and personal messaging are the most requested features. The table below shows the relevant results from the requirements questionnaire filled out by all the partners at the beginning of the year.

#### ▼ Requirements from questionnaire

Question	Germany	Prague	Lugano	Leeds	Madrid
Will you want your students to discuss work in groups?	Yes	No	Yes	Yes	Yes
How large do you expect those groups to be	20	1	20	3-5	30
Will you need tutor-moderated discussion?	Maybe	No	No	Yes	Yes
Will you need private contact with students?	No	No	No	Yes	Yes
Will you need synchronous	No	No	No	No	Yes

discussion ie instant chat?					
Will students need to access documents in the VLE?	Yes	Yes	Yes	Yes	Yes
Will you need asynchronous discussion (conference or bulletin board type discussion)?	Yes	Yes	Yes	Yes	Yes
Will you need inter-staff communication?	Yes	No	No	Yes	Yes
Will these documents be for Discussion? Downloading? Reading online? Printing? Other?	DI, DO, R, P	DO	DO, R, P	DI, DO, R, P, O (collaboratively constructed between students)	DO, R,P

From these requirements, we can say that the edukalibre system must provide:

- ◆ Group based discussions
- ◆ Tutor moderation of discussion groups
- ◆ Private messaging between users
- ◆ Document access for downloading, discussing, printing, reading, editing

Moodle, the VLE that will be adopted in Edukalibre, provides much of this functionality in its bulletin boards, file upload areas etc. However, even though Moodle claims to be based on constructivism, there are significant constraints on the way group work and discussions are supported. Moodle does not provide a coherent unified groupware application that pulls together these all individual groupware components a holistic manner. This significantly affects usability. Moreover, as Luca Bottury from the Lugano team commented, from the teacher's perspective Moodle is poor on supporting collaboration and its greatest flaw is the lack of effective group support.

For example, it may be desirable for users of the system to:

- ◆ Link to a document during a discussion
- ◆ Send a student a private message with a link to a relevant discussion
- ◆ Start a private chat with another student studying the same questions
- ◆ See which other users are online, and engage with them in collaborative learning

It is therefore proposed that a unified groupware application is required, which integrates well into moodle, and provides the functionality required by the partners in an easy to access and highly usable web application.

### ▼ 3 Functional specification

The system will be produced as an installable module for the moodle VLE, and will rely on an installation of Moodle for much of it's functionality.

Each section of the system will be linked to a central clipboard for each user. Any file or discussion message may be added to the clipboard. This central clipboard will enable people to easily refer to other objects when in any area of the system. For example, a user could view the resources for a particular subject, add a file reference to the clipboard, and then go to the discussions page, and paste a link to the file into a conversation.

This specification has been discussed with the creators of moodle, and with the moodle developer community.

The moodle community are very supportive of the idea, and would like to see some of the functionality integrated into the moodle core. It has been decided that whilst we could re-use a lot of the moodle database structure and code for discussion boards and resources, it would allow more flexibility to develop these simple components in house. The groupware module will be designed to replace the existing forum and resource modules of moodle. Ideas from the groupware module can then be integrated into the moodle core release should the developers see fit.

#### ▼ *Components to be produced*



The system will consist of the following components:

- ◆ **A database which will support the application**
  - ◆ Simple relational design
  - ◆ Tables integrated into moodle database
- ◆ **A document/file handling component**
  - ◆ Stores ANY type of file
  - ◆ Displays major picture/movie/document formats inline (with links to download)
  - ◆ Supports copying file references to clipboard
- ◆ **A messaging component**
  - ◆ Allows basic HTML messaging to any other user
  - ◆ Supports forwarding, replying, multiple recipients
  - ◆ Enables pasting of clipboard items into messages
  - ◆ Supports mailing to a moodle group (E.G teacher-to-class)
- ◆ **A discussion board component**
  - ◆ Single level of threading
  - ◆ Supports composing, replying and quoted replying
  - ◆ Supports copying message references to clipboard
  - ◆ Supports pasting of clipboard items into messages
  - ◆ Supports tutor moderation
  - ◆ Supports bookmark dropping and messages since last login

These sections will be integrated together in order to provide the functionality discussed above, and will integrate well into the existing moodle database, following the [coding conventions](#) and data layout recommended by the moodle developers.

### ▼ **3.1 Languages and Technologies**

#### ▼ **3.1.1 Web programming language**

The system will be written in PHP, which is the native language of moodle. php code will be written according to

Moodle's own coding guidelines for layout, commenting and variable naming. JavaDoc will also be used to comment the files.

### ▼ 3.1.2 Database system

The database will consist of a number of tables which will be added to the existing moodle database. The current installations we are using run under the MySQL database, but the resulting groupware application will install on systems running either MySQL or PostgreSQL. Database table and field definitions will conform to ANSI SQL standards, and should therefore be supported under the majority of database systems.

### ▼ 3.1.3 User interface language

The the output format or all browser display markup will be XHTML 1.0 strict. Tools are in place to validate this. All output produced by the groupware application will conform with thw WAI's AA guidelines.

The style and layout of the interface will be controlled with CSS 1.0, which supports the elements needed to provide a usable layout. CSS 2.0 is quite desirable for this application, due to it's more advanced layout handling and extra accessibility features, but many browsers still do not support CSS 2.0 properly, and in fact there are still elements of CSS 1.0 which Microsoft Internet Explorer for windows does not correctly render.

## ▼ 3.2 Documentation

### ▼ 3.2.1 User documentation

User documentation for the system will be produced in DOCBOOK format, and supplied to the users in both HTML(single page) and PDF format.

User documentation for this application will consist of:

- ◆ Interface tutorials with screenshots for each component
- ◆ Instructions for installing the groupware module
- ◆ Netiquette guidelines for social interaction online

### ▼ 3.2.2 Developer documentation

Developer documentation for the system will be produced in DOCBOOK format, and supplied to the users in both HTML(single page) and PDF format.

Developer documentation for this application will consist of:

- ◆ A structural tutorial of the layout and methodology used to create the application
- ◆ An overview of each of the php files that run the system
- ◆ API documentation for the classes and functions
- ◆ Automatic documentation through the use of JavaDOC commenting

## ▼ 4 Project timescale

The basic sections of work are: (deadlines in brackets)

### ▼ 4.1 Functional specification (31/03/04)

- ◆ Decide on the level of integration with the collaborative content editor
- ◆ Define the programming languages, database requirements, and output format of the system (PHP,SQL, XHTML/CSS)
- ◆ Define what information will come from the moodle database, e.g, user accounts
- ◆ Define the relational structure of the information across the whole system (moodle and groupware)
- ◆ Define the database structure which will support the application
- ◆ Define a basic OO framework (classes and functions) for the system.
- ◆ Define the interface requirements

- ◆ Define the testing methods, and the target goals

#### ▼ 4.2 Module prototype development (30/05/04)

- ◆ Create a basic moodle module and commit to CVS
- ◆ Create the database which will support the application
- ◆ Integrate the database into moodle
- ◆ Create the clipboard functionality
- ◆ Create the document storage section
- ◆ Create the messaging section
- ◆ Create the discussion board section
- ◆ Begin internal testing

#### ▼ 4.3 Module testing (25/06/04)

- ◆ Create the testing conditions
- ◆ Create testing documentation
- ◆ Run first set of tests
- ◆ Write testing report

#### ▼ 4.4 Prototype refinement and further testing(22/09/04)

- ◆ Refinement of prototype
- ◆ Prototype testing with partners
- ◆ Create developer documentation
- ◆ Create user documentation

#### ▼ 4.5 Delivery of module (30/09/04)

- ◆ Update developer documentation
- ◆ Update user documentation
- ◆ Create end of project report
- ◆ Submit work package to partners
- ◆ Submit groupware module to moodle project
- ◆ Submit groupware module to sourceforge.net

### ▼ 5 Afterword

This document has been produced thanks to conversations with various project partners during February 2004. As can be seen from the project timescale above, the deadline for the final specification of the groupware application is the end of March 2004.

There is therefore plenty of time for further consultation and discussion between the partners, particularly in relation to the integration of this application into the collaborative content editor.

We would welcome any comments, suggestions or feedback about any aspect of this document.