EAZYTAGZ – AN ENVIRONMENT FOR BUILDING POWERFUL INTERACTIVE TEACHING PORTALS FOR NON-PROGRAMMERS

Frode Eika Sandnes[1]

Abstract *¾ Teaching material on the web often comprise of static web pages. Interactive content can be achieved using proprietary and costly software packages. Some educators develop their own interactive teaching portals relying on specialised programming skills and knowledge, writing programs, scripts or assembling systems from prefabricated modules. These systems are often time-consuming to develop, hard to maintain and difficult to extend and apply by colleagues. This paper presents the open source teaching portal construction environment EazyTagz that is released in the public domain. EazyTagz provides simple building blocks for building interactive teaching portals trough a set of custom hypertext markup tags. EazyTagz is designed for non-programmers and can be entered as HTML style tags directly into HTML files or more conveniently using WYSIWYG HTML editors. This enables non-computer literate instructors to create powerful websites by focusing of pedagogical content rather than technical finesse and functionality. EazyTagz is built using Java Servlets and Java Server Pages technology and is easy to expand because of its open modular component architecture. EazyTags provides a large palette of pedagogical tools such as: electronic publishing and automatic navigation support, discussion groups, assignment submission and computer assisted testing. EazyTagz enforces complete separation of presentation and application logic simplifying maintenance and expansion.*

Index Terms ¾ online material, teaching infrastructure, discussion groups, online testing, assignment submission.

WEB BASED TEACHING TECHNOLOGY

Current web-based teaching technology broadly falls into three categories – commercial systems, systems freely available in the public domain and custom-made teaching portals. Commercial systems, such as WebCT [11], classFronter [4] and Blackboard [1], have achieved wide acceptance amongst educators. These systems are generally easy to use. However, the usability comes at the expense of flexibility, and such system generally are built around quite rigid and outdated teaching models that do not necessarily coincide with current pedagogical ideas and thought. Also such systems cannot be left to run themselves and often require a significant amount of administrative resources.

Teaching frameworks released in the public domain often suffer from being either too specialised and hard to adopt, or the technical quality is poor. Either the frameworks are too complicated and time consuming to deploy or they are unstable, to a degree were it affects the students' learning.

The high costs and restrictions imposed by the commercial systems and the difficulty of adopting public domain systems forces some educators to construct web based teaching portals from scratch. The most basic, and yet quite effective portals, are built from a set of static HTML files. Such portals usually include lecture notes, course information, messages from the teacher to the students, assignments, sample solutions, past exam papers and links to other course-related resources on the Internet. Such portals can be created inexpensively through a shared faculty web server. Teacher can compose the online teaching material as hand coded html files, using latex and the latex2html utility, or WYSIWYG tools such as Word, FrontPage and DreamWeaver. More sophisticated teachers employs cascading style sheets to achieve a more uniform look throughout the portal and perhaps even some scripting, for example PHP, to create discussion forums and assignment submission. Experiences show that teachers enjoy this control once they master the tasks of generating HTML pages with hyperlinks and moving these files onto the server. However, more advanced teaching portals are harder to maintain, mostly because the HTML code (presentation code) is mixed with scripting code (content generation code). This paper describes an infrastructure for building powerful interactive teaching portals where content presentation is completely separated from content generation using Java custom taglib technology.

MODERN WEB TECHNOLOGY

The multi-tiered application architecture have become the de facto way of constructing interactive web-portals [8]. The three-tiered architecture is the most common, comprising of the presentation tier, the application logic tier and the database tier. The web browser makes up the presentation tier and provides an application independent view. The application tier is provided by the web server or application server that performs all the processing on behalf of the clients (students). The database tier is usually realised by a relational database. Many teaching portal modules follow this architecture. For example, discussion groups are often implemented with CGI PERL scripts or PHP scripts which makes up the application tier. The scripts generate HTML

[1] Frode Eika Sandnes, Oslo University College, Faculty of Engineering, Cort Adelers gate 30, N-0254 Oslo, Norway, frodes@iu.hio.no

code that is forwarded to the students' web browser. The content of the discussions are stored in a relational database such as MySQL.

A problem that haunts web portals is the mixture of presentation and application logic. Technologies such as ASP (Active Server Pages), CGI-Perl scripts and PHP mix content presentation and content generation. Typically the application comprise of HTML pages with embedded code or scripts. HTML code with embedded scripts are hard to read and even harder to maintain.

The java J2EE enterprise application platform contains mechanisms that address these problems [3, 8]. The J2EE platform provides JSP (Java Server Pages) custom tag libraries. A custom tag library is a selection of custom defined tags that are assigned custom defined functionality. A custom tag can be compared to a procedure call where the name of the tag provides the procedure name, the tag attributes provide procedure parameters and the tag produces output or publishes Javabeans that are analogous to the procedure return value.

The tag attributes can be used to set tag options and control tag behaviour. The published Javabeans can subsequently be used by other tags and pages.

Applications composed from custom tags are easier to read and maintain because the presentation only contains markup codes and the functionality is hidden by the custom tags. A custom tag can be illustrated as follows:

```
<html>
  <body>
    The time is <mytag:date/>
  </body>
</thml>
```

This tag sequence generates a HTML page that displays the current date. The custom tag library defines a mapping between the <date/> tag and the Java code that generates the actual date. This strategy allows the layout to be modified independently of the application logic and vice versa. Custom tags can also iterate over their body several times, for example:

```
<html>
  <body>
    <mytag:for from="1" to "10">
      Inside the foor loop.
    </mytag:for>
  </body>
</html>
```

This tag sequence results in a HTML document where the line "inside the loop" is being displayed 10 times.

Information is exchanged between tags, read and written by the means of Javabeans – Javabeans are information capsules with properties that can be read and or written.

It is easy for Java programmers to define these custom tag libraries. Further, custom tag libraries simplifies the job of adopting existing code and functionality. Special purpose custom tag libraries are sold commercially and some are provided as open source in the public domain, for example Jacarta Taglibs1.0 that contains a set of low-level tags that gives access to the Servlet API (Application Programmers Interface). Taglibs are developed for special problem domains such as finance, online newspapers, e-commerce etc. There are to the best of our knowledge currently no custom tag libraries for building pedagogical interactive learning portals available. This paper describes eazytagz – an infrastructure for building interactive teaching portals for non-programmers. The infrastructure consists of a set of custom tags that simplifies the job of constructing highly customised and expandable teaching material on the web.

THE EAZYTAGS CUSTOM TAG LIBRARY

The eazytagz library currently addresses four specific problem domains, namely publishing online material, discussion forums, electronic coursework submissions and online testing. Each module contains a selection of intuitive custom tags. The teacher that constructs the teaching portal is given total control of the look and layout of the portal itself, especially if the tags are used in conjunction with CSS cascading style sheets. There are many aesthetically pleasing style sheets available on the Internet.

PUBLISHING ONLINE MATERIAL

Most educators have a need to publish information on the web for the students to read. This simplifies the task of distributing the material and it is more cost-effective from the educator's point of view. Students are given more freedom as they do not have to be present at specific locations at specific times to collect the material in person. Published material can include lecture notes, examples, exercises, assignments and sample solutions to selected assignments and exercises.

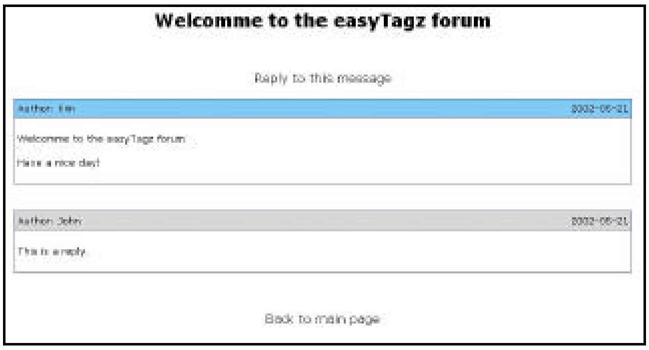
Published material usually follow a common pattern. There is often a navigation menu where students can select what to access or read and an area where the selected material is displayed. In fact, navigation is often a term used to describe both navigation and orientation. Orientation is the mechanism of knowing where the visitor is in the document hierarchy and navigation is the mechanism of moving from one location to another. One advantage of these patterns is that students are familiar with such structures and will immediate recognise and know how to use the navigation system. A pedagodical web site should always follow some layout conventions, and it should not take the student a long time to discover and learn how to naviagate the site. Poor portals have poor navigation support. Either the teacher has invented his or her own awkward navigation style, or the site hierarcy is poorly organised, and incompletely linked. A well organised and easily navigated web site is more fun and pedagogical to use and is more likely to stimulate learning compared to poorly organized web sites that makes students frustrated and disinterrested.

Either such frameworks are hand made static structures allowing the teacher a high degree of flexibility with more complex mainenance. Alternatively, the material can be stored in a database such that the navigation system can be generated dynamically on demand. Eazytags provides a combination of the two. The published files are simply placed in a directory and the navigation menu is generated automatically based on the files in the directory. The files can be a mixture of HTML, word, pdf documents and other media formats supported by the web browsers. A navigation menu is simply generated as follows using the loopLect tag and the lectureNodes tag.

```
<easyTagz:loopLect>
  <easyTagz:lectureNotes>
    <A HREF="</easyTagz:lectureNotes>
    </A>
    <BR>
</easyTagz:loopLect>
```

The loopLect is an iteration tag that traverses the list of documents and the start tag lectureNotes provides a description of the current document and the closing tag provides a hyperlink to the document. The teacher is free to add arbitrary formatting instructions around and between these tags. It is therefore easy to add the navigation bar vertically or horizontally, at the top or at the bottom, on the left or on the right and with text font in any colour size or shape. The advantage of this strategy is that new content is easily added by dropping the file to be published in the designated directory. Hyperlinks to the files are then automatically added to the navigation menus.

Comparatively, in a manual system the the teacher would have to update navigation hyperlinks whenever content is added or removed. This manual modification of hyperlinks is errorprone and time consuming. Further, it is also difficult to change the hand coded navigation style and design once the web site has been developed.



Welcomme to the easyTagz forum

Reply to this message

Author: Kim 2002-05-21

Welcomme to the easyTagz forum

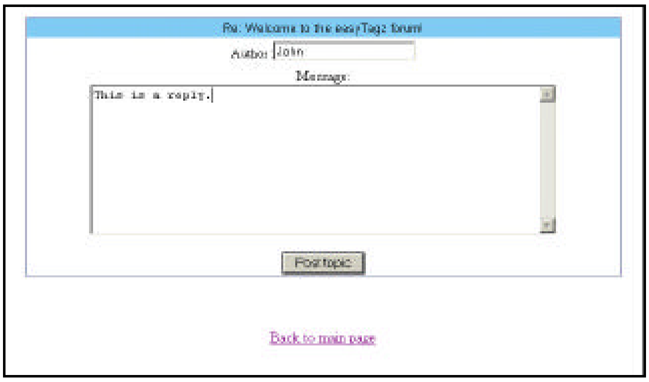
Have a nice day!

Author: John 2002-05-21

This is a reply

Back to main page

FIGURE 1:
OVERVIEW OF DISCUSSION GROUPS



Re: Welcome to the easyTagz forum!

Author: John

Message:

This is a reply.

FIGURE 2:
POSTING A MESSAGE TO THE DISCUSSION FORUM

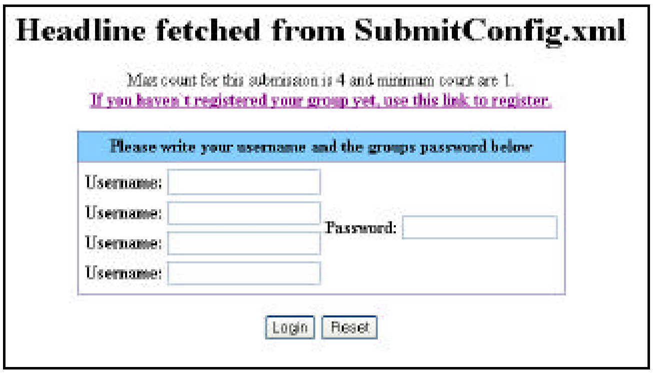
DISCUSSION FORUM

Many educators praise discussion forums and believe these are a great catalyst in the learning process as students collectively can collaborate and explore and exchange knowledge and ideas. A discussion forum or discussion group allows readers to post messages (see Figures 1 and 2). The messages contain a title and a textual message. Other readers can publicly reply the message with new postings and thus a discussion is created. Several different discussions can be underway simultaneously and this online discussion can often lead to very interesting exchange and development of ideas. Modern pedadogy promotes the interaction between students and learning by exchanging the students own previous experiences and knowledge. A discussion forum is also a great tool for the teacher as the teacher can monitor the activity on the discussion group and always know what issues are raised, and the teacher can interact too providing guidence or direct the discussions. The teacher can also pick up ideas for which issues to bring up in plenary sessions. Finally it can be used for course evaluation as the teacher can aquire hints about particular aspects of the teaching that does not work well.

There is a myriad of discussion forums available written in most flavours of web application languages such as ASP, PHP, CGI-perl/C etc. The advantage of the eazytags forum compared to most other forums is the ease of deployment. A forum can simply be added by placing the following sequency of tags in the presentation.

```
<jsp:useBean id="Db"
             class="easyTagz.Forum.DbBean"
             scope="application"/>

<easyTagz:LoopTag db="message">

  <jsp:getProperty name="Db"
                   property="messageId"/>

  <jsp:getProperty name="Db"
                   property="mdbid" />
```

```
    <jsp:getProperty name="Db"
                     property="mtopic"/>

    <jsp:getProperty name="Db"
                     property="mname"/>

    <jsp:getProperty name="Db"
                     property="ansvers"/>

  </easyTagz:LoopTag>
```



Headline fetched from SubmitConfig.xml

Max count for this submission is 4 and minimum count are 1.

If you haven't registered your group yet, use this link to register.

Please write your username and the groups password below

Username:

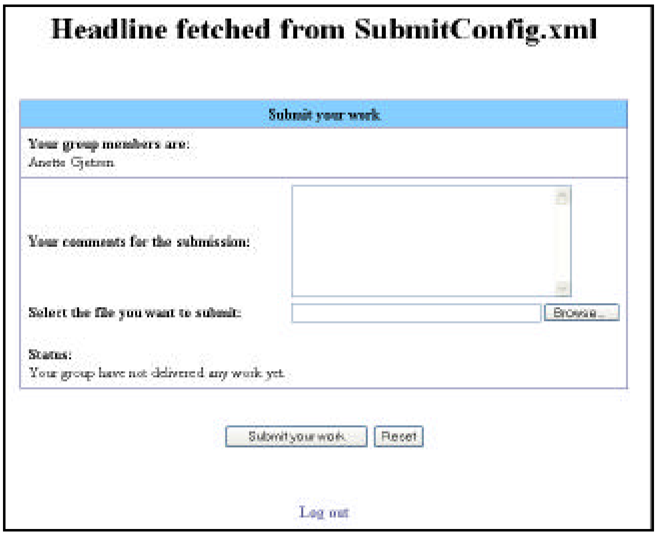
Username:

Username:

Username:

Password:

FIGUR 3:
AUTENTIFICATION FORM FOR GROUP SUBMISSIONS



Headline fetched from SubmitConfig.xml

Submit your work

Your group members are:
Anette Gjetnn

Your comments for the submission:

Select the file you want to submit:

Status:
Your group have not delivered any work yet.

FIGURE 4:
SUBMITTING COURSEWORK

There are no restrictions on how the custom tag libraries are used. For example, a page may contain several discussion forums, the forum can be a part of the page header, footer or in the left or the right document margin etc. The current version of eazytagz stores the discussions in a MySql relational database. MySql is popular amongst educators due to the public licence. However, the component-based Javabeans architecture simplifies the job of changing to a different storage method without affecting the rest of the system. Alternative storage methods can be another database product, an XML file etc.

The current implementation does not adress security issues such as content moderation and flame wars. Further, the administration of the discussion forum is currently carried out in the relational database by the database administrator.

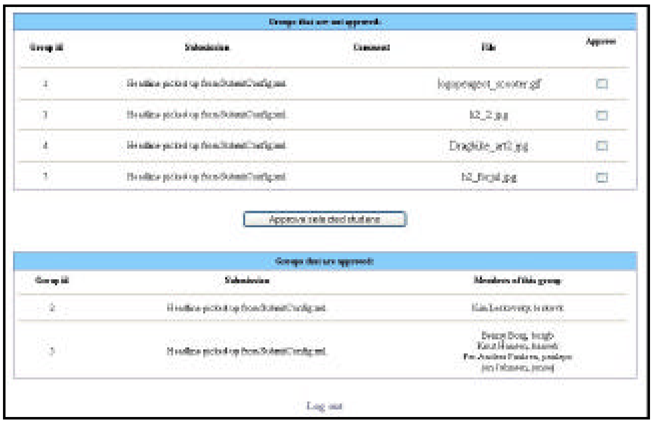
ELECTRONIC COURSEWORK SUBMISSION

The electronic coursework submission tool is more a managerial aid than an pedagogical aid. With large class sizes comprising of 30 to several hundred students it is time-consuming and laborious for the teacher to administer submitted coursework [9]. Traditional paper is easy handle, but the students submissions must then be accounted by hand which is laborious, time consuming and error-prone and not a task teachers enjoy. It is thus better to use teacher resources making the learning environment more pedagogical and improve the conditions for learning than wasting time bookkeeping and aministrating. Further, electronically submitted material is particularly beneficial in computer science related classes if the instructor is to test submitted program code etc. E-mail is a popular option but really not suitable for larger class sizes as discussed in [7]. Web based submission systems are a better option. Such systems consist of forms that allow students to authenticate themselves and submit the completed coursework. Electronically submitted material comes either in the shape of textual prose, hyperlinks to web sites where the coursework is located or simply files or compressed archives containing various kinds of documents, source code, binary code drawings etc. The eazytagz submission system allows individuals to submit work or groups using an interface inspired by a prototype developed by Burgess [2] (see Figures 3 and 4).

The uploaded files are placed in a specified directory, allowing teachers to easily inspect the coursework afterwards. Later, the students can log in and resubmit the files, and the existing file is overwritten when a new version is submitted. Students can also download previously submitted work for reference, or if they wish to make alterations. Finally, a student logging into the submission system will get an overview of the identity of the other group members, if the students are working in a group.

The submission system also has a validity attribute that is used to specify the time interval when the submission system is open. This mechanism prevents students from handing in coursework early, possibly before the assignment text has been published and eliminates the possibility of submitting coursework late. Everyone benefits from a strict deadline regime – teachers time is often wasted handling late submission or submissions not conforming to the guidelines.

The teacher can at any time view a list of submissions and can view the individual submissions. Each submission is associated with a designated checkbox. The teacher ticks this checkbox to accept the coursework. The coursework is then marked as accepted in the database. Students immediately see that their coursework is accepted if they log into the system (see Figure 5).



Groups that are not approved:

Group id	Submission	Comment	File	Approve
1	Headline picked up from SubmitConfig.xml		logoprosjekt_scooter.gif	☐
3	Headline picked up from SubmitConfig.xml		h2_2.jpg	☐
4	Headline picked up from SubmitConfig.xml		Dragkile_art2.jpg	☐
5	Headline picked up from SubmitConfig.xml		h2_[illegible].jpg	☐

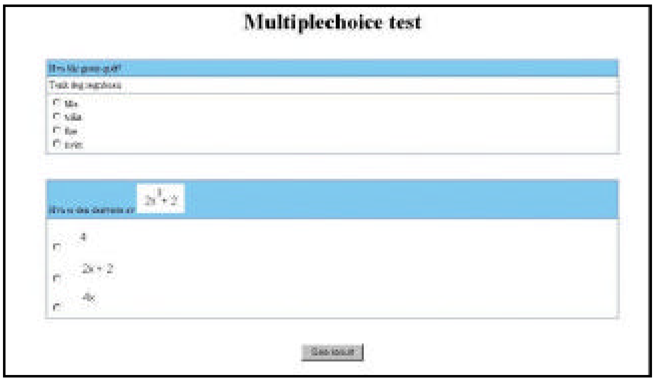
Approve selected students

Groups that are approved:

Group id	Submission	Members of this group
2	Headline picked up from SubmitConfig.xml	[illegible]
3	Headline picked up from SubmitConfig.xml	[illegible] [illegible] [illegible] [illegible]

Log out

FIGUR 5:
Teacher managing the submitted coursework.



Multiplechoice test

Hva blir grønn+gul?

[illegible]

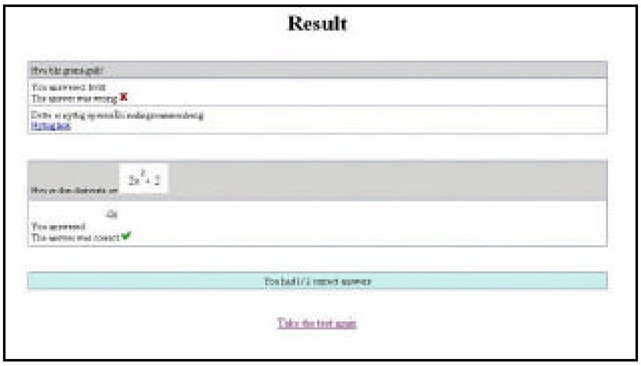
○ lilla
○ blå
○ fire
○ hvit

Hva er den deriverte av $2x^2+2$

○ 4
○ $2x+2$
○ $4x$

See result

FIGURE 6:
ONLINE MULTIPLE CHOICE TEST



Result

Hva blir grønn+gul?

You answered: hvit
The answer was wrong ✗

[illegible]
Nyttig link

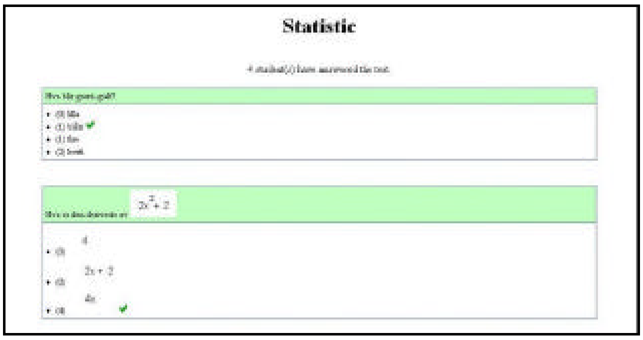
Hva er den deriverte av $2x^2+2$

$4x$
You answered:
The answer was correct ✓

You had 1/2 correct answers

Take the test again

FIGURE 7:
STUDENTS REVIEWING THEIR PERFORMANCE IN THE TEST



Statistic

4 student(s) have answered the test.

Hva blir grønn+gul?

- (0) lilla
- (1) blå ✓
- (1) fire
- (2) hvit

Hva er den deriverte av $2x^2+2$

- (0) 4
- (1) $2x+2$
- (3) $4x$ ✓

FIGURE 8:
TEACHER REVIEWING STUDENTS' PERFORMANCE IN THE ONLINE MULTIPLE CHOICE TEST

ONLINE TESTING

Educators in the engineering disciplines are obsessed with tests and electronic tests have been around for a long time. Still, there are no standard frameworks for implementing online tests in educational portals. Also newer literature on pedagogy [6] and activity groups [10] suggests that there has been an overfocus on testing and examination and that other forms of testing such as peer review and continous feedback is more effective in stimulating the learning process. Further, electronic testing is indeed quite a difficult issue. For example, where is the data to be stored? How long should the data be stored? How and when should answers be removed? How do one ensure security? Eazytagz does not attempt to solve all these problems, but provides a simple framework implementing simple multiple-choice self-test for students to review their own progress (see Figures 6, 7 and 8). Tests can be specified in a number of ways, for example using XML files with an intuitive syntax. These files contains a set of tags that specify question texts and the reply alternatives. Questions and answers can also be represented using images. The following XML extract demonstrates the multiple-choice specification:

```
<multiplechoice>
  <set>
    <question>
      What do you get if you mix green
      and yellow?
    </question>

    <wrong>purple</wrong>
    <correct>blue</correct>
    <wrong>four</wrong>
    <wrong>white</wrong>

    <link string="Useful link">
      http://regnbuen.no
    </link>

    <commentquestion>
      Think of the rainbow
    </commentquestion>
    <commentanswer>
      This is handy when painting
    </commentanswer>
  </set>
</multiplechoice>
```

A questionnaire can be embedded into any HTML page of the teaching portal as follows:

```
<form method="post"
      action="Result.jsp">
  <easyTagz:multiLoop type="question">
    <jsp:getProperty
       name="questionBean"
       property="questionString"/>
    <easyTagz:multiLoop type="alternative">
      <input
         type="radio"
         name="question<jsp:getProperty
           name="questionBean"
           property="questionCounter"/>"
         value="alternative<jsp:getProperty
           name="questionBean"
           property="alternativeCounter"/>"
      />
      <jsp:getProperty
        name="questionBean"
        property="alternativeString"/>
   </easyTagz:multiLoop>
 <input type="submit"
        name="finished"
        value="See result"/>
</form>
```

Tags are provided for building pages that show statistics. The statistics reveal the number of students answering particular questions and the percentage of replies for different alternatives.

With some programming knowledge it is relatively easy to generate very sophisticated tests. Question beans can be used to generate questions on the fly. The component-based Javabeans architecture allows new question generators to be plugged into the system without affecting or changing the other parts of the portal. Such a test generator could for instance generate multiplication questions for mathematics quizzes.

FUTURE DEVELOPMENT

This first version of eazytags is a prototype and is likely to go through several releases before it reaches commercial quality. However, such systems need to mature over time as users of the system provides feedback and new requirements and weaknesses are discovered. The most immediate issues on the agenda is to streamline and rework the tag naming and tag structure. It is also desirable to employ the built in autentification and access-control-list (ACL) feature of the JSP server rather than the program spesific autentification mechanism. This will simplify the program and administration. Further down the line it is desirable to employ enterprise database beans that provides a higher level of database independence, removing database specific code from the system altogether, completely relying on the enterprise container for data persistence. It is also possible to add a web interface to the system providing more high level approach to the teaching portal construction process more similar to existing commercial systems to satisfy users that are more comfortable adopting prefabricated templates.

SUMMARY

This paper presents the JSP custom taglibs for building flexible and interactive teaching portals for non-programmers. The system provides a set of tags with HTML/XML style syntax that is easier for non-programmers to relate to than traditional programming scripts. The tags are extremely powerful providing useful functionality. Details are hidden from the presentation and the teaching portals are thus more maintainable.

ACKNOWLEDGEMENT

The author would like to thank Anette Gjertsen, Kim Ophus Leskovsky, Per Anders Paulsen og Nicholas Ny Pettersen for the enthusiasm and hard work they have put into implementing the teaching portal framework and for providing the examples.

REFERENCES

[1] Blackboard product web site http://www.blackboard.com

[2] Mark Burgess, "Security and online student evaluation with large classes" in *Information technology and learning in engineering education*, HIO report series, 2002.

[3] Callaway, R. C. *"Inisde Servlets 2nd edition – server side programming for the java platform"*, Addision Wesley, 2001.

[4] Classfronter product web site http://www.fronter.com

[5] Fran H-L Jian and Frode Eika Sandnes, *"Written group assignments – experiences with collaborative web spaces"*, Proceedings of the ICEE International Conference on Engineering education, Manchester, United Kingdom, august, 2002 – submitted.

[6] Lauvås, Jakobsen "*Exit examen – eller? : Former for summativ evaluering i høgere utdanning*", Cappelen, 2002.

[7] Frode Eika Sandnes, *"Experiences with peer-review evaluation in computer science courses"*, Proceedings of the ICEE International Conference on Engineering education, Manchester, United Kingdom, august, 2002 – submitted

[8] Frode Eika Sandnes, *"Modern application development in Java for web: thin clients and fat servers"*, Tapir Press, August, 2002.

[9] Frode Eika Sandnes and Francine Hua-Li Jian, *"Quantitative Web-Based Teaching Tools for Progress Management and Evaluation"*, International Conference on Engineering Education, Oslo, Norway, pp 6E7-1-6E7-6, August, 2001.

[10] Students Against Testing, www.nomoretests.com

[11] WebCT product web site, http://www.webct.com