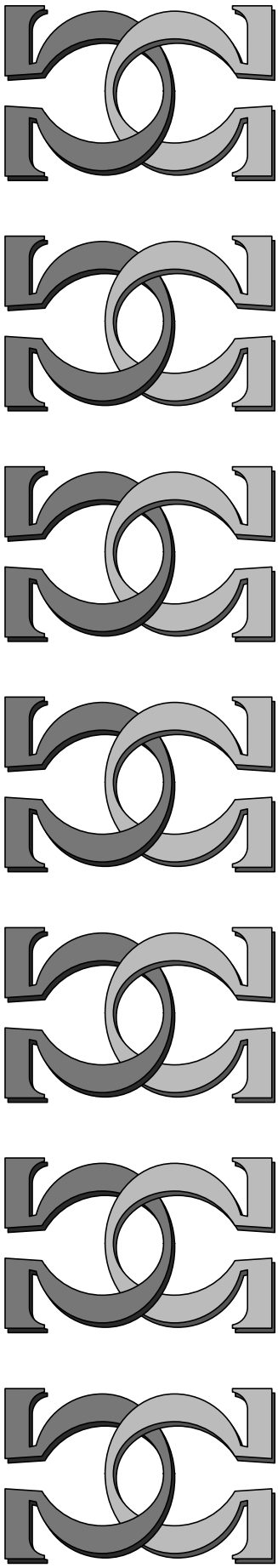

CDMTCS Research Report Series

A Conference Submission Web Server

So Young Park Lee
Michael J. Dinneen
Department of Computer Science
Univeristy of Auckland
Auckland, New Zealand

CDMTCS-157
June 2001

Centre for Discrete Mathematics and
Theoretical Computer Science

A Conference Submission Web Server

So Young Park Lee[1] with Michael J. Dinneen[2]
Department of Computer Science
University of Auckland
Private Bag 92019
Auckland, New Zealand

Abstract

We present a conference submission web server that supports conference organizers to manage the process of receiving and evaluating conference papers. The main features of our conference submission web server include effective operations and transactions, simple and flexible user interfaces, platform independence and reusability. The system models a conference organized by the Center for Discrete Mathematics and Theoretical Computer Science (Auckland, New Zealand). It is designed and implemented as a standalone application. It can be, however easily plugged into an existing conference web site.

1 Introduction

The Center for Discrete Mathematics and Theoretical Computer Science (CDMTCS) organizes conferences regularly. A conference web site is constructed and all the information about a conference is provided online in a web site. The web site also supports submissions of papers electronically. This report presents a new web-based application that supports an electronic submission for a conference as well as conference organizer's selection of accepted papers.

Previous electronic submission was made via e-mail. In such a system, a program committee member receives submitted papers in his electronic mail box. After receiving e-mail, he saves the file and records details about the submission. All the related jobs have been done manually. Another complication in the system is that an author has to send different messages; submission, resubmission or withdrawal etc., thus the committee member makes different actions according to the type of messages. A new system is required to help conference program committee members manage the process of receiving and evaluating papers while it still allows the authors to submit or resubmit papers. The aim of this project is to design and implement a conference submission web server, based on popular open source technologies.

[1] spar047@ec.auckland.ac.nz
[2] mjd@cs.auckland.ac.nz

The rest of this paper introduces this new system, which we call the "CCSS (**C**DMTCS **C**onference **S**ubmission **S**erver)". Section 2 provides the requirements analysis of the project. A user's guide is presented in Section 3. Design and implementation issues are discussed in Section 4 and finally, Section 5 concludes the paper and outlines some future plans.

2 Requirements Analysis

2.1 Functional Requirements

Our requirements analysis will start focusing on the different categories of users who interact with the system. In the CCSS, there are three different categories of users. The main activities of each user are as follows:

- *Author* submits a paper. And, once an author made a submission then he is allowed to revisit the system in order to resubmit a revised paper, to change his details or to withdraw the paper submission.

- *Conference chair* browses submitted papers, assigns papers to referees, browses referees' reports, evaluates papers according to referees' reports and finally accepts or rejects papers for the conference.

- *Referee* submits a referee report.

The CCSS should support all these activities. Several functional requirements include the following:

- A paper submission form is to be provided for an author to enter details of submission and to select a file on his machine.

- A resubmission form is to be provided for an author to resubmit a paper, change the details and withdraw the paper.

- The CCSS has to provide confirmation to the author after receiving the submission. A confirmation message is made on the screen as well as by email.

- A referee report submission form is provided for a referee to enter details of submission.

- The CCSS should provide a user interface for the conference chair to help browsing papers and referee's reports.

- A file should be uploaded into the server machine's file system and it can be downloaded when it is requested.

- A login process is required that only authorized people access secured information.

- Information about papers and reports should be stored in a secure place and they should be recoverable in case of system failure.

- The CCSS should be pluggable in an existing conference web site, of which provides information and structures of the conference.

- Simple update mechanisms for conference chair should be provided to reuse the CCSS for another conference.

- The user interface should be simple and easy to use.

2.2 Software Requirements

The CCSS is designed and implemented by using open source technologies. The basic idea behind open source is very simple, "optimal exchange of knowledge". When programmers can read, redistribute, and modify the source code for a piece of software, the software evolves. People improve it, people adapt it and people fix bugs [8]. This rapid evolutionary process produces better software than the traditional closed model, in which only a very few programmers can see the source and everybody else must blindly use an opaque block of bits.

For CDMTCS, the CCSS application is run on a Linux machine that provides high performance for dynamic web application. In this section we briefly describe the features of the software used for the CCSS.

2.2.1 MySQL Database

MySQL is the most popular Open Source SQL database management system. MySQL is a relational database management system highly suited for accessing databases on the Internet. MySQL has following advantages:

- It is Open Source Software, which means that it is possible for anyone to use and modify. Anybody can download MySQL from the Internet and use it without paying anything as long as it isn't for commercial purposes. Anybody so inclined can study the source code and change it to fit their needs.

- MySQL offers a rich and very useful set of functions. It is a client/server system that consists of a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and several programming interfaces. It is very fast and easy to use.

- MySQL is reliable. It uses security based on Access Control Lists (ACLs) for all connections, queries, and other operations that a user may attempt to perform. There

is also some support for SSL-encrypted connections between MySQL clients and servers [10].

- It has a lot of contributed software available. Many applications or languages already support it, including PHP.

2.2.2 Apache Web Server

Apache has been the most popular web server on the Internet since April of 1996. More than half of the web sites are believed to use Apache. Apache has the following notable features:

- Apache provides cross-platform support. It runs on Windows (95/98/NT), OS/2, and all the major variants of Unix.

- Apache supports various protocols. The server is fully compliant with HTTP/1.1 and supports API and ISAPI (NT).

- Apache is a very secure web server. The SSH protocol is used to secure these connections with strong encryption, which provides a tunnel between the two communicating machines [9].

- Apache's attractiveness also lies in its extensibility and its freely distributed source.

2.2.3 PHP Development Tool

PHP, which stands for "PHP: Hypertext Preprocessor", is a server-side HTML-embedded scripting language that allows developers to create dynamic web pages. PHP-enabled web pages are treated just like regular HTML pages and you can create and edit them the same way you normally create regular HTML pages. It is a powerful language and the interpreter, which can do everything that CGI can do; it is able to access files, execute commands and open network connections on the server.

PHP is favorable over several other similar scripting languages:

- PHP's features are comparable to Microsoft's ASP (Active Server Pages) and VBScript. Both PHP and ASP have effective APIs for building database-driven applications. However, unlike ASP, PHP is open source and cross-platform. PHP can run on Windows NT under Microsoft's IIS web server or on any UNIX variant as an Apache module or CGI.

- One of its major attractions over Perl and JavaScript is its seamless IP connectivity, with LDAP and TCP as well as the IMAP mail interface.

- Perl does all that PHP can do, but a lot of people find Perl more difficult to learn and a little too top-heavy. Also, mod_perl can be a bit imposing.

One of notable features of PHP is its support for a wide range of databases; it supports mSQL, MySQL, dBase, Oracle, Ingres, Sybase, ODBC, Informix and many more [1]. However, PHP isn't just a database-centric language. It does so much more: dynamic graphics generation, IMAP, SNMP, LDAP and XML support.

PHP is a rich and full-featured language that provides numerous functions. With PHP, MySQL and Apache Web Server, creating *dynamic database-driven* web server is not a difficult task.

3 A User's Guide

This section provides a user's guide for the CCSS. To set up CCSS for a conference, users are required to configure the global variables and to create a database of tables. Section 3.1 explains how to configure global variables for a particular conference and Section 3.2 provides an instruction to create the database.

The CCSS consists of four different subsystems. Each subsystem offers users different operations and transactions. A guide of the subsystems is presented in Section 3.3.

3.1 Define Global Variables

All the global variables are defined only once in the "globals.inc.php" and this modular component is used for the entire application. There are several global variables that should be defined for a particular conference:

Conference title
$conference_title[3]**:** the title of the conference,
e.g., "Discrete Mathematics and Theoretical Computer Science 2001".

Variables that are used to send email to author confirming the submission
$conference_email_addr: the mailer(conference committee) 's email address,
e.g., "abc@cs.auckland.ac.nz".
$return_path: Return path for emailing errors.
$message: Extra information about the conference. The contents of this message variable are attached to the submission confirmation email.

Variables that are used to upload files
$upload_path: the directory name, in which the files are stored,
e.g., "/home/conference/files/"
$symlink_path: the directory name, in which the symbolic links are stored,
e.g., "/home/conference/file_links/"

[3] Variables in PHP like many other scripting languages, must have a $ in front of their names.

Variables that are used for connection to MySQL database
$hostname: the host name, e.g., "localhost".
$username: the user name
$password: the password

3.2 Create Tables in MySQL Database

In order to use the CCSS, a MySQL database has to be set up and two tables are required in the database, called *papers* and *reports*. Two PHP programs are provided for creating them. "CreateSubmissionDatabase.php" creates a new database called *Submission*, and "CreateTables.php" creates two tables in the database. The specification of these tables follows.

Field	Type	Description
lgnID(primary key, auto increment)	Int	The unique paper identifier.
paperKey	varchar(10)	The unique encrypt key value for each submitted paper. The value is used as the paper ID.
paperTitle	varchar (80)	The full title of the paper.
athNameList	varchar (80)	Multiple authors' names.
authorName	varchar (48)	The full name of the contact author.
firstNames	varchar (38)	The first names of the contact author that are extracted from full name.
lastName	varchar (10)	The last name of contact author that is extracted from the full name. It is used to sort papers.
email	varchar (25)	The email address of the contact author.
fileType	varchar (5)	The submitted file type, must be "pdf" or "ps".
status	int (1)	The status of the paper. It is set to 0 by default. not known = 0, accepted = 1, rejected = 3, withdrawn = 2
date	Date	The last submission date.
time	Time	The last submission time.

Table 1: Description of fields in *papers* table.

Field	Type	Description
reportID (primary key, auto increment)	Int	The unique referee's report ID over all submitted reports.
paperID	Int	The id of the paper of which the referee reviewed.
originality	int (1)	The originality. (1 for no result to 5 for very original)
quality	int (1)	The technical quality. (1 for low to 5 for high)
relevance	int (1)	The relevance to the conference. (1 for out of scope to 5 for very relevant)

clarity	int (1)	The clarity of presentation. (1 for poor to 5 for Excellent)
recommendation	int (1)	The overall recommendation. (1 for reject to 5 to accept)
refreeName	varchar (48)	The full name of the referee.
confidence	varchar (6)	The referee's confidence on his evaluation. ("Medium", "High" or "Expert")
justification	Text	The justification of evaluation.
comments	Text	The comments for the authors.
date	Date	The report submission creation date.
time	Time	The report submission creation time.

Table 2: Description of fields in *reports* table.

3.3 A User's Guide of CCSS Subsystems

The CCSS can be divided into 4 different subsystems, called *Submission System*, *Resubmission System*, *Referee Report Submission System* and *Paper Presentation System*. In this section, we describe these subsystems and present a user's guide of each system. Figure 1 presents the system components of the CCSS.

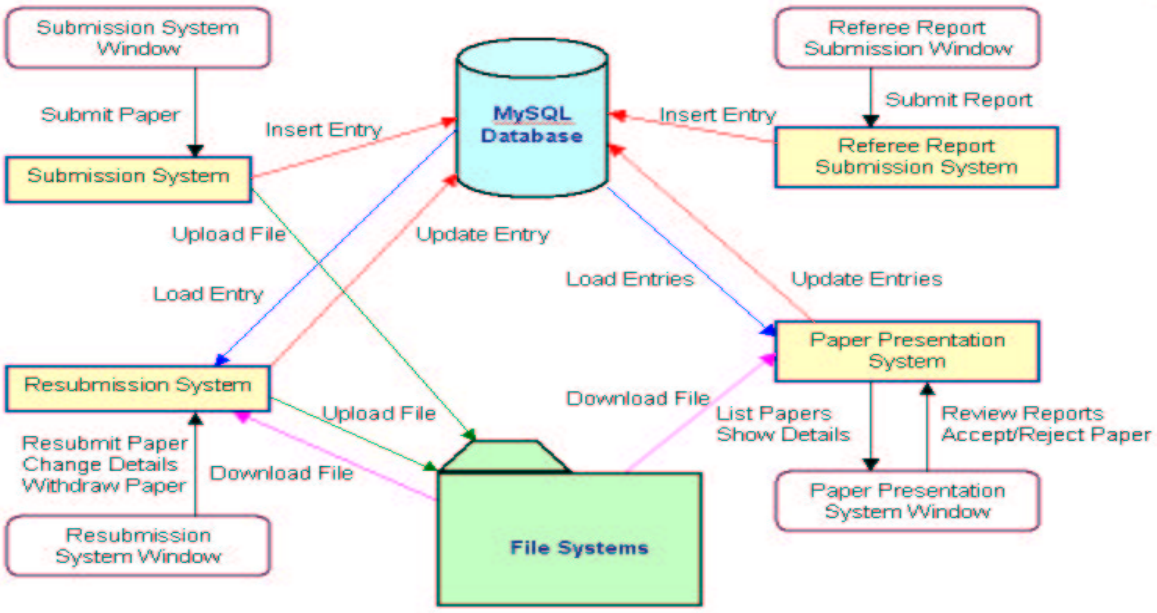

Figure 1: Components of the CCSS.

3.3.1 Submission System

The Submission System allows authors to create new submissions. The paper submission form is displayed for an author to fill in. The form fields include:

- Paper Title – the paper title.
- Author List – the list of the author names.
- Contact Author Name – the name of the contact author. It allows two different name formats. If the first name is "Sophia" and the last name is "Lee" then both "Sophia Lee" and "Lee, Sophia" are allowed.
- Contact Email – the email address of the contact author.
- Contact Email (for confirmation) – this should be identical to the email address input above.
- File to submit – the full path and file name of a file. The author can press browse button to select a file that automatically makes fill in this field.

Each contact author can select a file from his machine and the CCSS uploads the file into the server machine's file directory. Figure 2 shows the screenshot of the paper submission form of the Paper Submission System window, which is plugged within a conference web site. It shows that an error message is displayed when an author does not provide two identical email addresses.

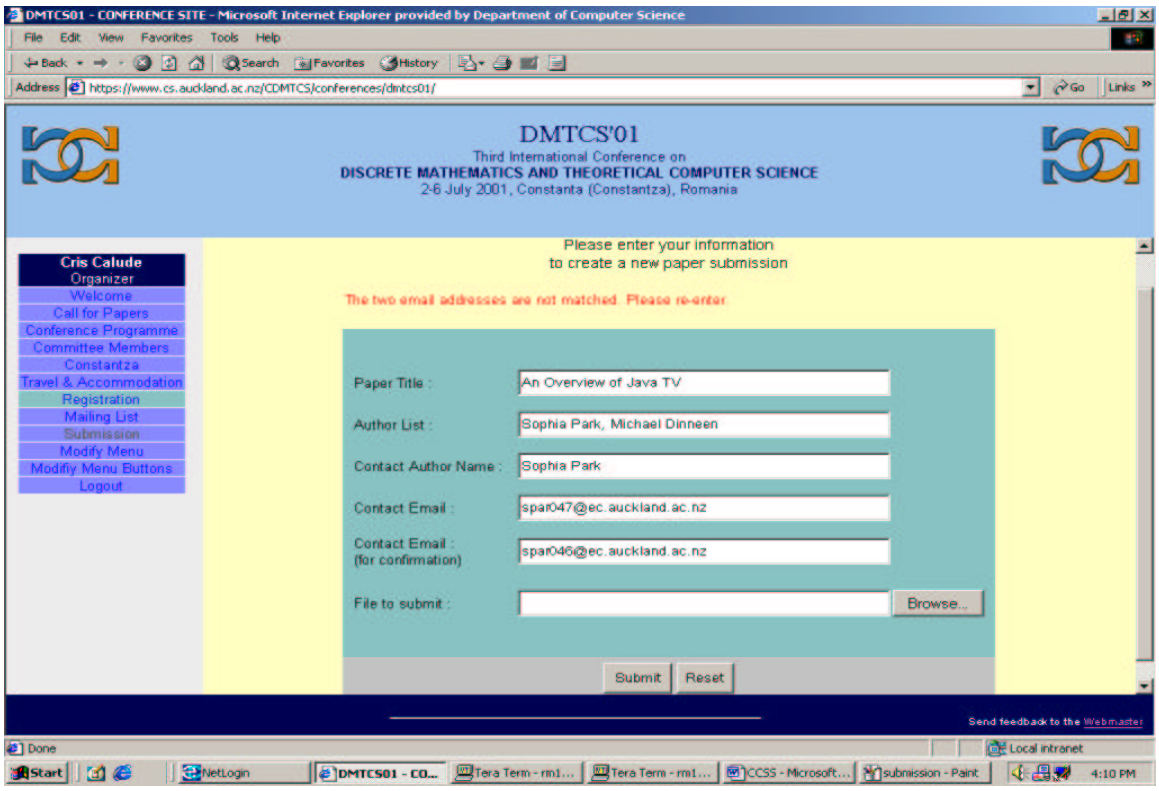

Figure 2: Screenshot of the Submission System Window.

After all the input fields are validated the submission is accepted and stored in the database. The CCSS gives a unique paper ID for an author to present this paper ID when he uses the Resubmission System. A confirmation message is made on the screen as well as by email.

3.3.2 Resubmission System

The Resubmission System supports an author's ability to resubmit a revised paper, to change details of his submission or to withdraw the previously submitted paper. The author should login in order to access the information kept in the database. Figure 3 shows a screenshot of the login entry window of the Resubmission System.

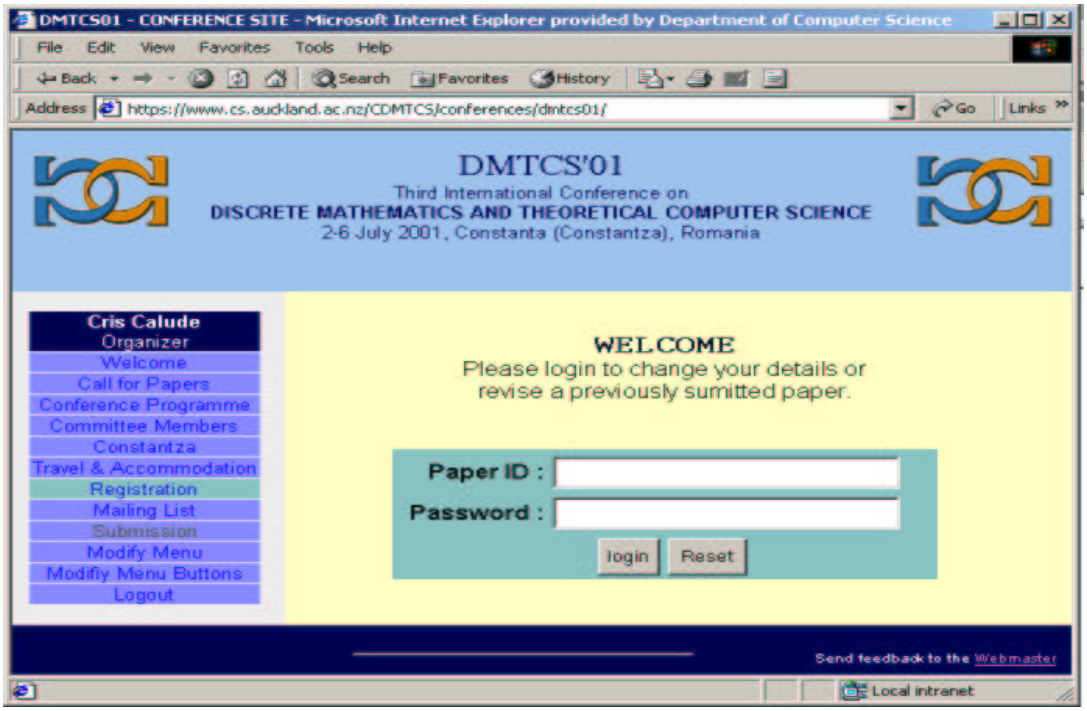

Figure 3: Screenshot of the Resubmission System Window to login.

The author is required to enter a paper ID and his email address as his password. After they are validated, details of his previous submission are displayed with 3 different option buttons; change details, resubmit paper and withdraw paper.

- *Change details* option allows the user to change the paper title, the authors' list, the contact author name and the contact author's email address. The new details are displayed after changes are made.

- *Resubmit paper* option provides a file selection form, which allows the user to select a file and resubmit it for the conference.

- *Withdraw paper* option allows the user to withdraw previously submitted paper. The confirmation message is displayed to protect a mistake.

Each user may choose one option at a time and, in each case, can cancel it. The user can also view his previously submitted paper by pressing the paper title. Figure 4 shows the screenshot of the Resubmission System Window that an author can change the details of previous submission.

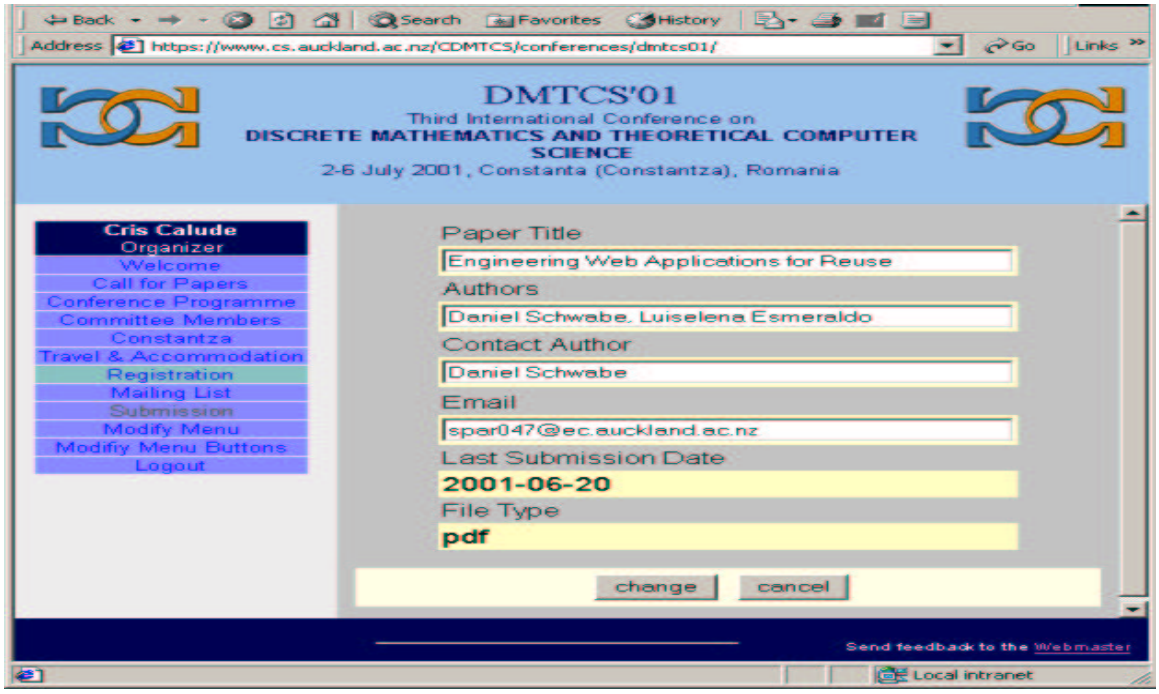

Figure 4: Screenshot of the Resubmission System Window to change details.

3.3.3 Referee Report Submission System

The Referee Report Submission System offers referees to submit their reports on line. A referee should present correct paper ID that he evaluated. Figure 5 presents a screenshot of the referee report submission system window.

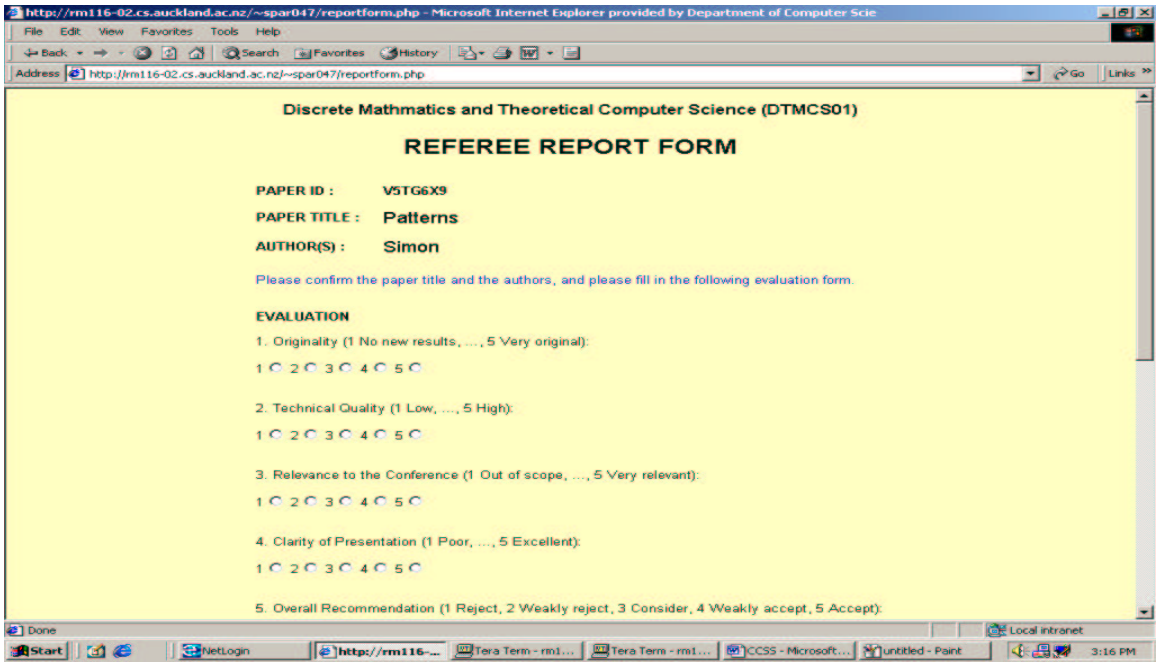

Figure 5: Screenshot of the Referee Report Submission System Window.

Evaluation categories are originality, technical quality, relevance to the conference, clarity of presentation and overall recommendation. Each referee chooses at most one option from 1 to 5 for these questions. Text input fields are also provided for the referee to write his justification of evaluation and comments.

3.3.4 Paper Presentation System

The Paper Presentation System supports the activities of the conference chair and program committee. When he opens the paper presentation window, a list of submitted papers is available. Several list options are provided for the conference chair to choose a subset of the list to view. The options are:

- *Submitted* option lists all submitted papers, which includes accepted, rejected, withdrawn, and not known papers. This is default option.
- *Accepted* option lists all accepted papers.
- *Rejected* option lists all rejected papers.
- *Not known* option lists all papers, which is submitted but not withdrawn and neither accepted nor rejected yet.
- *Withdrawn* option lists all withdrawn (by author) papers.

Another option box is provided to list the papers ordered by paper number, contact author's last name and last submission date. Figure 6 shows the screenshot of the Paper Presentation System, where shows the list of accepted papers is ordered by author.

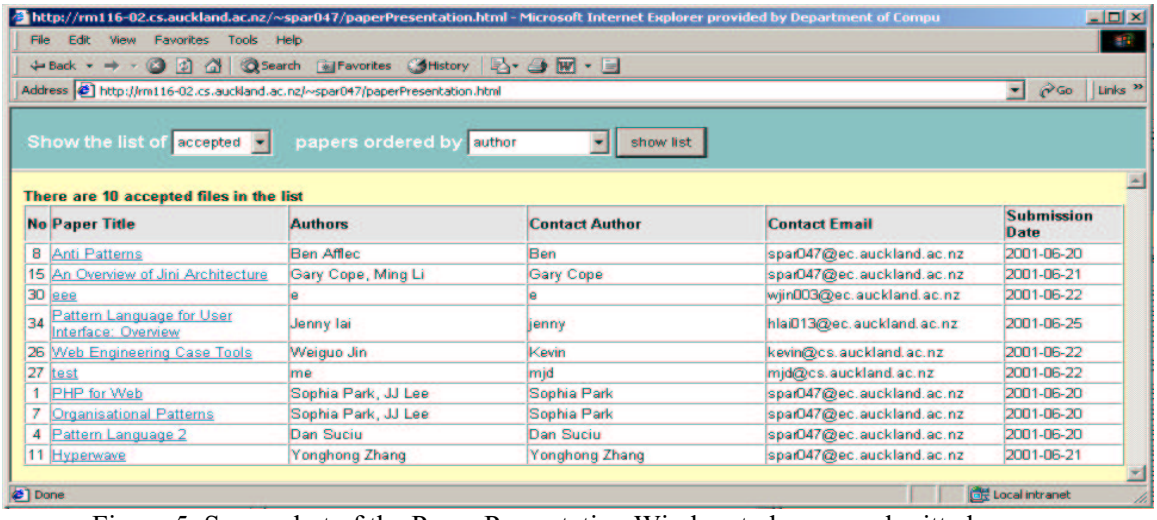

Figure 5: Screenshot of the Paper Presentation Window to browse submitted papers.

The details of each paper are displayed by clicking on a paper title. This paper-details page provides the following operations for the conference chair.

- A link to the file is provided. When the conference chair presses on the "Show the paper", new window is opened and the contents of the file are shown on the new window.

- The conference chair can view the referees' reports about the paper by clicking the "show" button at the bottom of the page. "Hide" button is also offered to hide the report details.

- "Accept" and "reject" buttons are provided for conference chair to select papers for the conference.

Figure 6 presents a screenshot of a paper-details page of the Paper Presentation System.

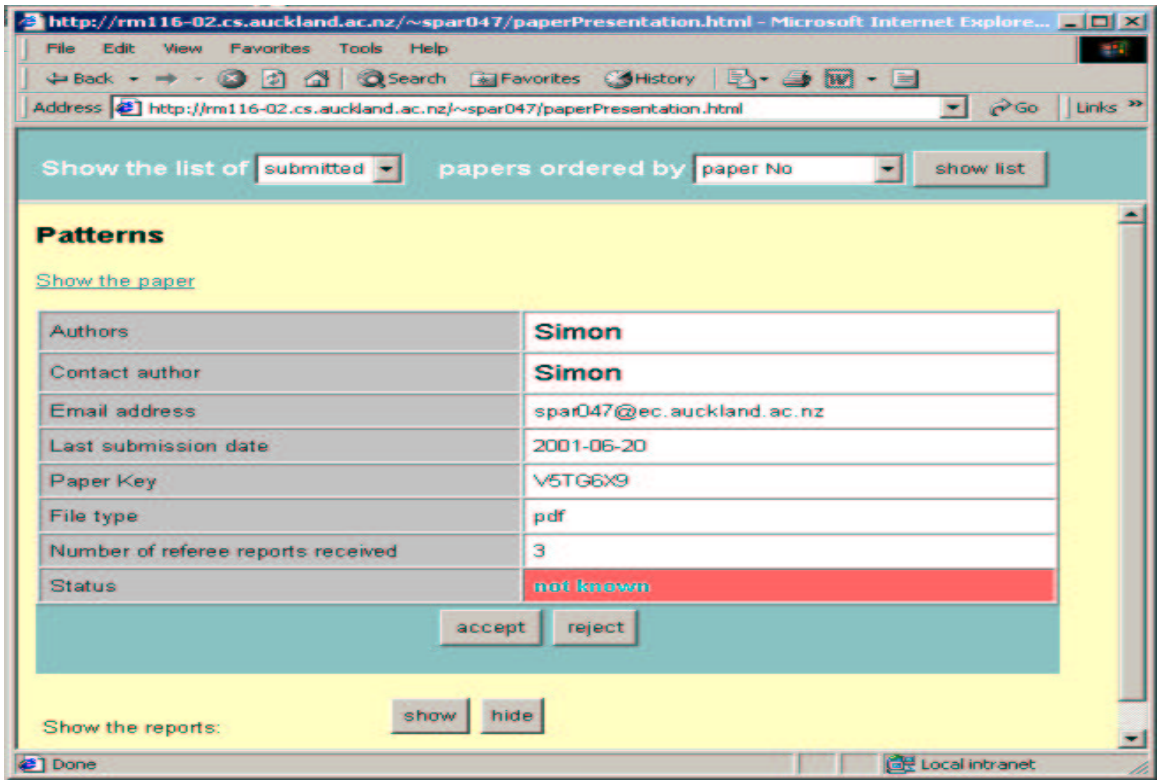

Figure 6: Screenshot of the Paper Presentation Window to view submitted paper details.

4 Design and Implementation Issues

This section explains some design and implementation issues in CCSS. In Section 4.1 specifications of PHP scripts for the CCSS are presented. The file uploading routines are introduced in Section 4.2. And we discuss the usages of encrypted paper keys in the CCSS and also describe how to generate it in Section 4.3. Finally Section 4.4 explains email functions in the CCSS.

4.1 PHP Script Specification

PHP scripts are used to implement CCSS. PHP is a rich and full-featured language that provides numerous functions. This section describes the algorithm of PHP scripts of the CCSS subsystems in Section 4.1.1 to 4.1.4. It also explains the modular components used in the CCSS in Section 4.1.5.

4.1.1 submission.php

This script implements the Submission System. It firstly shows a submission form on the browser. It allows a user to select a file from his machine. When the user clicks the submit button, followings are checked.

- Are all the input fields entered?
- Are two email addresses identical?
- Is the email address correctly formatted?
- Does the domain name of email address exist?

If any condition does not meet then an error message are displayed on the browser. When an error message is shown, previous input field values are remained the same so that the user does not need to enter all again. After validation of the fields, the script loads information into "paper" table in the database. MySQL database server gives unique paper number. Using it, an encrypted paper key for a submitted paper is generated. The usage and how to generate this paper key is discussed in Section 4.3.

A selected file is uploaded and the paper key becomes its new file name in the CCSS file system. If an error occurs during loading an entry into database or uploading a file, the submission is cancelled and an error message is displayed on the browser. After processing submission details, the confirmation is made on the browser as well as via email.

4.1.2 resubmission.php

This script implements the Resubmission System. It firstly shows a login form on the browser. A paper ID and contact author's email address are entered, and the submit button is clicked on. The script compares each field with stored attribute value of the paper respectively. If they are all matched then the user is displayed previous submission details on the browser with three option buttons as explained in Section 3.3.2. The paper title is a link to the file in CCSS file directory. The script does following for each option:

- When the user selects the *Change details* option, it results in updating entry of the paper in "papers" table in the "Submission" database.

- When the user selects the *Resubmit paper* option, it uploads selected file and replaces it with previously submitted file. The last submission date is updated accordingly.

- When the user selects the *Withdraw paper* option, it does not result in deletion of the entry in "papers" table but it changes in status of the paper.

- When the user presses on the paper title, it downloads the file from the CCSS file directory and opens a new window to show this file.

4.1.3 reportform.php

These script implements Referee Report Submission System. Firstly a referee is required to enter a paper ID to view a full report form. When referee clicks the submit button the script checks if all the fields are entered. Error messages are displayed if any of mandatory fields is missing. Referee does not need to fill all the fields again since previously entered fields remain the same on the browser. MySQL database server gives a unique report id and the fields are loaded in the *report* table in the database.

4.1.4 menus.php, list.php, showDetails.inc.php and reportDetails.inc.php

These are scripts to implement the Paper Presentation System. The "menus.php" displays paper list options and order options on a browser and waits for a user's input. When user selects an option from each option box and clicks "show List" button then it passes these options to "list.php" to handle. The "list.php" loads all the entries from the "papers" table in the database according to the selected options and presents them on the browser. Each paper title is a link to a paper-details page.

The "showDetails.inc.php" displays the paper-details page. It offers the conference chair following operations:

- When he presses on the "Show the paper" link, it downloads the file from the CCSS file directory and opens a new window to display the file.

- When he clicks the *Accept* button, it updates the status of the paper to "accepted". And, when he clicks on the *Reject* button, it updates the status of the paper to "rejected".

- When he clicks the *Show* button to view the reports, it loads all the entries from the *reports* table that is associated with the particular paper, and presents them. The user is also provided the *Hide* button to hide the reports' details.

4.1.5 globals.inc.php, db_mysql.inc.php and styleSheet.inc.php

These are modular components for the CCSS. Modular components are small, self-contained objects that make up a bigger object [1]. Using them makes the CCSS more maintainable because they have only one copy for whole system. All of these items can be turned into a single module and referenced within the rest of the CCSS. Each serves a particular purpose as below:

- “Globals.inc.php” allows a user to configure the global variables for a particular conference. The details of this script are presented in Section 3.1.

- “Db_mysql.inc.php” is implemented the database transaction functions.

- “StyleSheet.inc.php” is the style sheet for the CCSS user interfaces.

4.2 File Uploading

File uploading is one of the powerful features in the CCSS. A user can select a file to submit from his machine and the CCSS uploads it into its file directory. PHP is capable of receiving file uploads from any RFC-1867 compliant browser (which includes Netscape Navigator 3 or later, Microsoft Internet Explorer 3 with a patch from Microsoft, or later without a patch) [6]. This feature lets a web developer upload both text and binary files. The following file upload form is used in the Paper Submission System to receive the details of the submission and to allow the author to select a file.

```
<form ENCTYPE="multipart/form-data" NAME="Submission" ACTION="submission.php"
METHOD="post">
        Paper Title :
        <input TYPE="text" NAME="paperTitle" VALUE=”” MAXLENGTH="80" SIZE="48">
        Author List :
        <input TYPE="text" NAME="athNameList" VALUE="" MAXLENGTH="80" SIZE="48">
        Contact Author Name :
        <input TYPE="text" NAME="authorName" VALUE="" MAXLENGTH="48" SIZE="48">
        Contact Email :
        <input TYPE="text" NAME="email" VALUE="" MAXLENGTH="48" SIZE="48">
        Contact Email (for confirmation) :
        <input TYPE="text" NAME="email2" VALUE="" MAXLENGTH="48" SIZE="48">
        <input type="hidden" NAME="MAX_FILE_SIZE" VALUE="1000000">
        File to submit :
        <input TYPE="file" NAME="userfile" VALUE="" MAXLENGTH="48" SIZE="48">
        <input TYPE="submit" NAME="submit" VALUE="Submit">
        <input TYPE="submit" NAME="submit" VALUE="Reset">
</form>
```

“Submission.php” receives the fields and handles them. Two fields called “MAX_FILE_SIZE” and “userfile” are used for file uploading.

- The “MAX_FILE_SIZE” hidden field must precede the file input field and its value is the maximum filesize accepted. The value is in bytes.

- The “userfile” is the temporary filename in which the uploaded file is stored in the temporary directory on the server machine.

All the other input fields are loaded into *papers* table and we do not discuss further about it. The following PHP function is used to upload selected file.

```
function doUpload( $userfile, $id, $type) {
        if ( is_uploaded_file ( $userfile ) ) {
                $encryptkey = "paper".$id;
                $enc_key = encrypt ( $encryptkey );
                $newfilename = $enc_key.$id;
                $path = $GLOBALS[ "upload_path" ];
                $upload_file = $path.$newfilename;
                $upload_file.=".";
                $upload_file.=$type;
                copy( $userfile, $upload_file );
        }else{
                $newfilename = "";
        }
        return $newfilename;
}
```

This function takes three parameters; an uploaded file in temporary directory ($userfile), unique paper number ($id) which is given by MySQL database server and a file type of the file ($type). *Is_uploaded_file (String userfile)* is PHP predefined function that returns true if the file named by userfile was uploaded via HTTP POST. If it returns false then the *doUpload* function returns "" to indicate an error. Otherwise, a new file name ($newfilename) is given and it is copied into CCSS file directory. The file will be deleted from the temporary directory. The function finally returns a new file name to store it in database.

4.3 Encrypted Paper Key and File Security

One important issue in the CCSS is its file security. A file in CCSS file directory must not be accessed by anyone except its author or a conference organizer. Thus the CCSS generates unique encrypted paper keys for submitted papers. This paper key basically has following three functions:

- It is used as a file name when the selected file is uploaded into CCSS file directory. It ensures that nobody can guess a file name in the CCSS file system.

- It is also used as a login paper ID when the author revisits the conference web site to use resubmission system.

- It is also used as a login paper ID when a referee submits a referee report about this paper.

An encrypted paper key consists of two parts. The first six characters are generated by the CCSS encryption mechanism. The unique paper number that has been generated by the MySQL database server is added. For example, "V5TGx619" is a file name of which

"V5TGx6" is an encrypted string, and last two digit "19" represents unique paper number. The following function is used to generate the paper key:

```
function encrypt ( $string ) {
// encrypt $string, and store it in $enc_text
  $result = gensalt ( );  // generate salt
  $enc_text = crypt ( $string, $result );
  $filtered_text = "";
  $i = 0;
  $count = 0;
  do{
    $c = substr( $enc_text, $i, 1);
    if( ( ord( $c ) >= ord( "0" ) ) && ( ord( $c ) <= ord( "Z" ) ) ){
            $filtered_text .= $c;
            $count++;
    }
    $i++;
  } while( $count < 6 || $i == strlen( $enc_text ) );
  if( strlen ( $filtered_text ) < 6 ) {
            do{
              $filtered_text .= random_chr();
            } while( strlen ( $filtered_text )  < 6 );
  }
  return $filtered_text;
}
```

The function firstly generates an encrypted string using the standard Unix DES encryption method. From this encrypted string, first six printable characters are extracted. If we get less than 6 printable characters then we add random characters to make up a six-character string. Then unique id is added at the end of it so that whole file name is unique.

Now, files in the CCSS file system are not easily guessable by others. However, a conference organizer may require these files directly from the server; he needs to get hard copies of files using a command line on the server machine. It is required to provide easy and effective file names for the conference organizer to help access the files in the CCSS file system. Thus we provide another file directory only for the conference organizers and create symbolic links to all the submitted files in the directory.[4] The symbolic links have the specified names that are explicitly shown the unique paper number. The symbolic links are named "paper" and its unique paper number such as "paper1"or "paper25".

[4] This symbolic links do not work on the Windows systems.

4.4 Email Functions

The CCSS offers some useful email functions. It can check whether an email address is correctly formatted and it verifies the existence of the domain associated with the address. The following *validateEmail* function is used in the Paper Submission System to do this task.

```
function validateEmail ( $email ) {
  // check email format firstly and it verifies the existence of the domain
   if ( eregi ("^[0-9a-z]([-_.]?[0-9a-z])*@[0-9a-z]([-.]?[0-9a-z])*\\.[a-z]{2,3}$", $email, $check)){
            if ( getmxrr ( substr ( strstr($check[0], '@'), 1), $validate_email_temp)){
               return "valid";
            }
            if ( checkdnsrr ( substr ( strstr ( $check[0], '@'),1),"ANY") ) {
            return "valid";
            }
     }
    return "not valid";
}
```

Two PHP predefined functions are used to verify the existence of the domain:

- *Getmxrr* searches DNS for MX records corresponding to a given Internet host name.

- *Checkdnsrr* checks DNS records corresponding to a given Internet host name or IP address. It catches DNSs that are not MX.

It also sends an email confirmation when it receives a submission. The *mail($to, $subject, $body, $from.$replyTo)* automatically mails the message specified in *body* to the receiver specified in *to*. The fourth string argument is inserted at the end of the header. We specifies the mailer (*from*) and the return address for any mailing error (*replyTo*). These parameters are set as explained in Section 3.1.

5 Conclusions and Future Work

This project introduced CCSS, a new CDMTCS Conference Submission Server. It has following features:

- It consists of four different subsystems. Each subsystem supports a different kind of users in a conference to accomplish their tasks. Whole or part of the CCSS can be easily plugged into an existing conference web site.

- Each subsystem provides a user-friendly interface on a browser that gives a user a great flexibility. And it provides basic capabilities of function which include input field entry operations, input field error checking, various database transactions and reset mode.

- Each subsystem is implemented as a separate module so that it can be used separately. And furthermore, the CCSS uses several modular components that crosscut multiple PHP scripts. These modular components include a global variable configuration, a style sheet and a database operations script. It makes the CCSS very flexible and maintainable.

- The CCSS ensures the security of its conference submission data and files by its unique authentication and encryption mechanism.

The CCSS provides many powerful operations for the process of a conference. Several improvements can be made on the CCSS for even more powerful and dynamic conference web site:

- The CCSS can be integrated with the existing CDMTCS conference web site to provide more dynamic visibility and different capabilities of operation for a different category of users.
    - Allowing a user to use a login name and a password for the whole conference web site. Thus the user can view different information of the conference and have different capabilities of operation according to his category.

- More functions can be added on Paper Presentation System:
    - Searching function can be added so that the conference chair can easily find a particular paper.
    - Different views can be provided for the conference chair to be able to have a view he wants.
    - Statistics about the conference can be provided; authors nationality.

References

[1] Christopher Cosentino. *Essential PHP for web professionals,* Prentice Hall, 2001.

[2] Luciano Baresi, Franca Garzotto, and Paolo Paolini. *Extending UML for Modeling Web Applications,* 0-77695-0981-9/01 2001 IEEE.

[3] Daniel Schwabe, Luiselena Esmeraldo, Gustavo Rossi and Fernando Lyardet. *Engineering Web Applications for Reuse*, 1070-986X/01 2001 IEEE.

[4] Engin Kirda, Mehdi Jazayeri, Clemens Kerer, and Markus schranz. *Experiences in Engineering Flexible Web Services,* 1070-986X/01 2001 IEEE.

[5] *What can PHP do?* http://www.php.net/manual/en/intro-whatcando.php

[6] *Handling file uploads* http://www.php.net/manual/en/features.file-upload.php

[7] *Using remote files* http://www.php.net/manual/en/features.remote-files.php

[8] *Open Source Approach*, www.openSource.org

[9] *About Apache* http://www.apache.org/ABOUT_APACHE.html

[10] *An Introduction to MySQL* http://www.mysql.com/articles/mysql_intro.html

[11] *Building a Database-Driven Web Site Using PHP and My SQL* http://www.mysql.com/articles/ddws/index.html