ABSTRACT

This project is a comprehensive application for conducting online tests for Physics students enrolled in courses managed by the Texas Electronic Coalition for Physics (TECP). Online Grading System (OGS) is developed for the Department of Physical and Life Sciences of Texas A&M University – Corpus Christi. OGS provides Web-based interface for instructors, students, TECP coordinators and chairperson. Five Universities in Texas will be offering online tests for physics courses to students through OGS. OGS provides dynamic generation of tests which have questions with variables in them. The values of the variables are unique for each student and are generated on the fly. Each question can have an image associated with it. The values on the image too can be generated dynamically. All the dynamically generated values for each student are stored for future reference. Solutions too are computed by the OGS dynamically depending on the random values and formula. The solution keyed in by the student is compared with the solution generated and the score and letter grade are assigned to each student.
# TABLE OF CONTENTS

ABSTRACT ................................................................. i  
TABLE OF CONTENTS ................................................ ii  
LIST OF FIGURES ...................................................... iii  
LIST OF TABLES ........................................................ iv  
1 INTRODUCTION AND BACKGROUND .................. 1  
2 NARRATIVE ............................................................ 2  
  2.1 TECP Chair Module ........................................ 3  
  2.2 System Administrator Module ....................... 5  
  2.3 Local Coordinators Module ......................... 5  
  2.4 Instructor Module ........................................ 9  
  2.5 Student Module ........................................... 16  
3 SYSTEM DESIGN .................................................... 19  
  3.1 System Specification ................................... 19  
    3.1.1 Server Configuration .......................... 19  
    3.1.2 Client Configuration ......................... 20  
  3.2 Overview of the System ............................. 20  
  3.3 Database Tables and Schemas .................... 22  
  3.4 Entity Relationship Diagrams .................. 29  
  3.5 Input/Output and Processing ................... 30  
  3.5 Protocols and Interfaces .......................... 31  
  3.6 File Formats ............................................. 32  
  3.7 Data Dictionary ........................................ 33  
4 IMPLEMENTATION .................................................. 34
4.1 Dynamic Generation of Values for Variables in a Question .................................. 34
4.2 Dynamic Generation of Solution for Each Question ........................................ 35
4.3 Dynamic Generation of Values on Images .................................................. 35

5 TESTING AND EVALUATION ................................................................. 37
5.1 Testing for Downloading Speed ............................................................ 37
5.2 Testing for Security of Data ................................................................. 38
5.3 Testing for Session Consistency ............................................................. 38
5.3.1 Unit testing ......................................................... 38
5.3.2 Acceptance testing ................................................................. 39
5.3.3 Stress testing ................................................................. 39
5.3.4 Regression testing ................................................................. 39

6 FUTUREWORK ..................................................................................... 40

7 RESULTS AND CONCLUSION ............................................................ 41

8 BIBLIOGRAPHY ...................................................................................... 42

9 APPENDICES ........................................................................................... 43
Appendix A Database Table Definitions.................................................. 43
Appendix B Source Code CD-ROM ............................................................. 50
<table>
<thead>
<tr>
<th>Figures</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 2.1 Coordinator page to enter instructor data</td>
<td>3</td>
</tr>
<tr>
<td>Figure 2.2 Entering course information related to instructor</td>
<td>4</td>
</tr>
<tr>
<td>Figure 2.3 Coordinator page to enter semester data</td>
<td>4</td>
</tr>
<tr>
<td>Figure 2.4 System administrator enters coordinator data</td>
<td>5</td>
</tr>
<tr>
<td>Figure 2.5 Coordinator password change page</td>
<td>6</td>
</tr>
<tr>
<td>Figure 2.6 Coordinator personal information change page</td>
<td>6</td>
</tr>
<tr>
<td>Figure 2.7 Research/Internship posting page</td>
<td>7</td>
</tr>
<tr>
<td>Figure 2.8 Student information update page</td>
<td>8</td>
</tr>
<tr>
<td>Figure 2.9 Authorize students to TECP</td>
<td>9</td>
</tr>
<tr>
<td>Figure 2.10 Instructor personal information change page</td>
<td>10</td>
</tr>
<tr>
<td>Figure 2.11 Add new student academic details</td>
<td>10</td>
</tr>
<tr>
<td>Figure 2.12 Modify an existing students academic details</td>
<td>11</td>
</tr>
<tr>
<td>Figure 2.13 File Transfer Protocol application for TECP</td>
<td>12</td>
</tr>
<tr>
<td>Figure 2.14 Add/Modify Assignment details</td>
<td>13</td>
</tr>
<tr>
<td>Figure 2.15 Create a new image for a question</td>
<td>13</td>
</tr>
<tr>
<td>Figure 2.16 Create a new question for a test</td>
<td>14</td>
</tr>
<tr>
<td>Figure 2.17 Post research/internship opportunity page</td>
<td>14</td>
</tr>
<tr>
<td>Figure 2.18 Delete research/internship opportunity</td>
<td>15</td>
</tr>
<tr>
<td>Figure 2.19 Student home page</td>
<td>16</td>
</tr>
<tr>
<td>Figure 2.20 Student Password change page</td>
<td>16</td>
</tr>
</tbody>
</table>
Figure 2.21 Student personal information change page ............... 17
Figure 2.22 Courses being offered .................................. 18
Figure 2.23 Internship and research opportunities .................. 18
Figure 3.1 Schematic diagram of the System ......................... 21
Figure 3.2 ER diagram of application ............................... 29
Figure 3.3 Diagrammatic view of data flow ......................... 30
Figure 3.4 Schematic diagram of protocols and interfaces ........ 31
Figure 3.5 Data Dictionary for OGS ................................. 33
LIST OF TABLES

<table>
<thead>
<tr>
<th>Tables</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 3.1 Server Configuration</td>
<td>19</td>
</tr>
<tr>
<td>Table 3.2 Client Configuration</td>
<td>20</td>
</tr>
</tbody>
</table>
1. INTRODUCTION AND BACKGROUND

Online Grading System is a Web based application to ease the process of offering classes, conducting quizzes, assigning assignments, and grading the assignments and tests.

Dr. Balasubramanya, Associate Professor of Physics at Texas A&M University – Corpus Christi (TAMU-CC) is interested in conducting the quizzes in physics courses using World Wide Web thus mitigating the labor of conducting them and grading them manually, but at the same time maintaining the same stringent standards for quiz administration as would apply in traditional quizzes. The proliferating pace at which the Internet is entering into the lives of all the people and the myriad conveniences it has to offer demands an online quiz program. The time and physical labor thus saved on the part of the instructor can be utilized efficiently and effectively in a more resourceful area thus ultimately leading to more productive work.

The whole application has been deployed on Dell 2600SC server running on Redhat Linux 8.0 operating system. Apache web server supports the PHP (Hypertext Pre Processor) scripts. MySQL Database Management System is used with PHP to store the databases on the server.
2. NARRATIVE

Online Grading System supports the following features such as creating student database through HTML (Hyper Text Markup Language) front end, creating courses, assignments, quizzes, and the tests database through HTML front end, file uploading and downloading from the courses and generation of tests. Both downloadable format for the quizzes and online format are being supported by the application.

To make this application unique from other existing applications a number of features have been added to the application. Following are a few features that are specific to this application:

- Dynamic (on the fly) generation of images for questions
- Variable substitution with random values in questions
- Downloadable (PDF) format generation of the tests
- Variable substitution with random values on images
- File Transfer Protocol

Following categories of people have been identified to interact with the application:

- TECP (Texas Electronic Coalition for Physics) chairperson
- System Administrator (Sysadmin)
- Local Coordinators
- Instructors
- Students
Additional features such as changing password and inserting or modifying address details have been implemented for all users. These features are accessible by all users and are similar in working. Apart from the common features each individual category of user has his/her own specific needs which have to be supported by the application. A consistent template has been adopted to give the user a familiar feeling of the application after using several times.

All the features and details of the features have been listed in the following sections:

2.1 TECP Chairperson Module

2.1.1 Enter instructor data

TECP chairperson is the user who gets the roster of all the instructors, the courses they are teaching and the semester the instructors are teaching those courses. Instructor ID, password and Home directory for each instructor will be entered through this form as shown in Figure 2.1.

![Figure 2.1 Coordinator page to enter instructor data](image)
2.1.2 Enter course/instructor data

TECP chairperson has to enter details which relate each instructor with the courses he/she is offering and the semester the Course will be offered and the corresponding figure is shown in Figure 2.2.

![Figure 2.2 Entering course information related to instructor](image)

2.1.3 Enter semester data

Each semester has specific beginning and ending dates which change almost every year. TECP chairperson has to enter details about the semester prior to the start of the semester. This is done through the form as shown in Figure 2.3. Semester ID, Beginning date and Ending date for the semester are entered through this feature. This information entry is essential for the proper working of the application.

![Figure 2.3 Coordinator page to enter semester data](image)
2.2 System Administrator Module

2.2.1 Enter coordinator data

Every institution has a local coordinator who takes care of the instructor and student issues at the institution. A friendly interface as shown in Figure 2.4 allows the System administrator to enter these details.

![Figure 2.4 System administrator enters coordinator data](image)

2.3 Local Coordinators

Local coordinator is responsible for all the activities at his/her institution level. Instructors of an institution report to local coordinator. Local coordinator enters the student data into the OGS application at the beginning of the semester from the roster.

2.3.1 Change password

The Sysadmin creates a user ID and password for the local coordinator, which the local coordinator can change after he/she logs in into the application with the assigned User ID and password. Figure 2.5 shows the form for this purpose.
2.3.2 Change personal information

Details such as Address, Phone number and FAX number can be changed by the Local coordinator after he/she logs into the application. Figure 2.6 shows the form for this purpose.
2.3.3 Post research/internship opportunities

Details related to available research opportunities or internship opportunities can be posted using this feature. An associated URL can be specified to make the opportunities available to all the users of the application. Figure 2.7 shows the interface for this purpose.

![Research/Internship Posting Page](image)

Figure 2.7 Research/internship posting page

2.3.4 Update student information

Details pertaining to a student can be changed by the local coordinator without being aware of the password of the student. As shown in Figure 2.8, this feature has been provided to change the details of the students after he/she leaves the university.
2.3.5 Authorize new TECP students

At the beginning of every semester the local coordinator gets the roster of the students who have registered for TECP courses. The local coordinator enters a student’s First name, Last name and the Course ID using this feature. The details added using this form makes a student eligible to login into the application and access features determined for him/her. Figure 2.9 shows the form for this purpose.
2.4 Instructors Module

2.4.1 Change password

The Instructor can login into the application and change his/her password to whatever he/she wants to.

2.4.2 Change personal information

As shown in Figure 2.10, details such as Address, Phone number, FAX number can be changed by the instructor after he/she logs in into the application. The instructor or the local coordinator can view this data.
2.4.3 Add/Modify student academic information

The instructor can use this feature to change the existing details of a test for a student in the selected course or enter the test details for a new student. Figure 2.11 and Figure 2.12 show the forms for this purpose.

Figure 2.11 Add new student academic details

Figure 2.10 Instructor personal information change page
2.4.4 Generate/View class statistics

Grades obtained by the students in a given class can be viewed using this feature. Information retrieved in this form can be arranged in ascending order based on student ID or name. The scores obtained by the students on individual tests and the final score are retrieved in this form.

2.4.5 Generate/View course report

An instructor can view the grades obtained by the students in a course he/she is offering using this feature. The results can be arranged in ascending order either by name or scores.
2.4.6 Import course content

This feature enables the instructor to upload, download, copy, move, delete and edit files on the server. The instructor may want to upload images related to test of a course or any other study material related to the courses he/she is teaching. A directory with the Course ID is created and all the files related to a course are stored in that directory. Figure 2.13 shows the corresponding form.

Figure 2.13 File Transfer Protocol application for TECP.

2.4.7 Add/Modify assignment details

This feature enables an instructor to create new assignments/quizzes or modify an existing assignment/quiz details. Assignment details such as percentage weight, posted date, end date and URL of the assignment and number of questions in an assignment can be changed using this feature. Figure 2.14 shows the form.
2.4.8 Create/Modify test

Create/Modify test feature enables an instructor to create a set of questions for a test, specify details of an image for a question, editing an existing question or image.

Figure 2.15 and Figure 2.16 show the form.
2.4.9 Post research/internship opportunity

Instructors can post research or internship opportunities on the server for students and other instructors and students to visit. The research/internship opportunities posted by the instructors and coordinators are accessible to all the instructors, coordinators, TECP chair and students registered. The corresponding forms for this feature are shown in Figure 2.17 and Figure 2.18.
Figure 2.18  Delete research/internship opportunity
2.5  Student Module

![Image of student home page](image)

Figure 2.19  Student home page

2.5.1  Change Password

Every student authorized to use the application can login with the User ID and password created by the local coordinator and then can change the password to whatever the student wants to. Figure 2.20 shows the corresponding form.

![Image of student password change page](image)

Figure 2.20  Student password change page

16
2.5.2 Add/Update Personal Information

Details such as address, phone number, cell number, phone number, email address, etc can be changed by the student after logging into the application. Figure 2.21 shows the corresponding form.

![Personal Information Form](image)

Figure 2.21 Student personal information change page

2.5.3 View Courses offered this semester

A student can view the list of TECP courses offered in the current semester. All the courses listed are with a hyperlink (optional) on which he/she clicks to view the details of the course. Details such as instructor, his/her contact number and syllabi, etc are available for the student to view. The corresponding form is shown in Figure 2.22.
2.5.4 Internship and research opportunities

A student can view the internship and research opportunities posted by instructors and local coordinators on this page. Figure 2.23 shows the corresponding form.
3. SYSTEM DESIGN

3.1 System Specification

System design covers the hardware configuration and software details of the Online Grading System. This chapter explains the interaction between the different components of the OGS. Database schemas, database design, ER diagrams, data flow diagrams, interfaces and protocols used to develop the OGS and various file formats used for the OGS are explained.

3.1.1 Server configuration

The server deploys the Online Grading System application for all clients to access it. Below is the server configuration. The details are shown in table 3.1.

Table 3.1 Server Configuration

<table>
<thead>
<tr>
<th>Machine</th>
<th>Dell Server 2600 SC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Redhat Linux 8.0</td>
</tr>
<tr>
<td>Web Server</td>
<td>Apache</td>
</tr>
<tr>
<td>Software</td>
<td>PHP</td>
</tr>
<tr>
<td>Database</td>
<td>MySQL</td>
</tr>
<tr>
<td>Other Accessories</td>
<td>Internet connection</td>
</tr>
</tbody>
</table>

3.1.2 Client configuration
The client is used to just interact with the server. Client machine can be any personal computer or Macintosh computer with availability of Internet and a browser.

Table 3.2 shows the details of the Client configuration

<table>
<thead>
<tr>
<th>Machine</th>
<th>Any computer which supports HTTP.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>Windows, Linux, Unix.</td>
</tr>
<tr>
<td>Browser</td>
<td>Any Web browser (such as Internet Explorer, Netscape)</td>
</tr>
<tr>
<td>Other Accessories</td>
<td>Internet connection.</td>
</tr>
</tbody>
</table>

3.2 Overview of the System

Figure 3.1 shows a schematic diagram of the Online Grading System. Major components of the OGS are:

The User Computer or Client System is the client-side computer. The requirements for a client-side computer are a PC or a Macintosh with a browser that supports Javascript. Examples of such browsers are Microsoft Internet Explorer, Netscape Navigator, Opera, Mozilla etc.

The Server is the Galileo server at Texas A&M University – Corpus Christi. Galileo server works on Redhat Linux 8.0 version.

The Web server is the Apache server installed on Galileo. Apache is an open source Web server which is used to support server side scripting languages such as PHP.
The request sent by the client to the Galileo server is sent to the Apache Web server which parses the PHP script and responds appropriately.

MySQL database gets the request internally from the Apache Web server if the request has a database query. The request so obtained is parsed by the SQL engine of the MySQL and the response is sent back to the Apache webserver.

LaTeX engine gets the request from the Apache with the result obtained from the MySQL engine. This request is parsed by the LaTeX engine and processed to generate PDF files. The final result obtained from the LaTeX engine is sent back to the client by the Apache Web server.

![Diagram of the system](image)

**Figure 3.1 Schematic diagram of the system**
3.3 Database Tables and Schemas

Following are the database tables that were used for the deployment of the OGS system:

### 3.3.1 Coordinator

This table contains the information of Coordinator. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>crd_id (PK)</td>
<td>Text</td>
<td>20</td>
<td>Coordinator ID</td>
</tr>
<tr>
<td>first_name</td>
<td>Text</td>
<td>20</td>
<td>First Name</td>
</tr>
<tr>
<td>middle_initial</td>
<td>Text</td>
<td>20</td>
<td>Middle Initial</td>
</tr>
<tr>
<td>last_name</td>
<td>Text</td>
<td>20</td>
<td>Last Name</td>
</tr>
<tr>
<td>pass_wd</td>
<td>Text</td>
<td>20</td>
<td>Password</td>
</tr>
<tr>
<td>sch_id</td>
<td>Text</td>
<td>8</td>
<td>School ID</td>
</tr>
<tr>
<td>email_id</td>
<td>Text</td>
<td>40</td>
<td>Email ID</td>
</tr>
<tr>
<td>phone_no</td>
<td>Text</td>
<td>12</td>
<td>Phone Number</td>
</tr>
<tr>
<td>fax_no</td>
<td>Text</td>
<td>12</td>
<td>Fax Number</td>
</tr>
<tr>
<td>address</td>
<td>Text</td>
<td>250</td>
<td>Address</td>
</tr>
</tbody>
</table>

### 3.3.2 Course

This table contains the information of Course. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>crs_id(PK)</td>
<td>Text</td>
<td>8</td>
<td>Course ID</td>
</tr>
<tr>
<td>crs_name</td>
<td>Text</td>
<td>40</td>
<td>Course Name</td>
</tr>
<tr>
<td>crs_desc</td>
<td>Text</td>
<td>250</td>
<td>Course Description</td>
</tr>
<tr>
<td>crs_dir</td>
<td>Text</td>
<td>40</td>
<td>Course Directory</td>
</tr>
</tbody>
</table>

### 3.3.3 Instructor

This table contains the information of Instructor. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ins_id (PK)</td>
<td>Text</td>
<td>20</td>
<td>Instructor ID</td>
</tr>
<tr>
<td>first_name</td>
<td>Text</td>
<td>20</td>
<td>First Name</td>
</tr>
<tr>
<td>middle_initial</td>
<td>Text</td>
<td>20</td>
<td>Middle Initial</td>
</tr>
<tr>
<td>last_name</td>
<td>Text</td>
<td>20</td>
<td>Last Name</td>
</tr>
<tr>
<td>Field Name</td>
<td>Type</td>
<td>Size</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------</td>
<td>------</td>
<td>----------------</td>
</tr>
<tr>
<td>pass_wd</td>
<td>Text</td>
<td>20</td>
<td>Password</td>
</tr>
<tr>
<td>sch_id</td>
<td>Text</td>
<td>8</td>
<td>School ID</td>
</tr>
<tr>
<td>email_id</td>
<td>Text</td>
<td>40</td>
<td>Email ID</td>
</tr>
<tr>
<td>phone_no</td>
<td>Text</td>
<td>12</td>
<td>Phone Number</td>
</tr>
<tr>
<td>fax_no</td>
<td>Text</td>
<td>12</td>
<td>Fax Number</td>
</tr>
<tr>
<td>address</td>
<td>Text</td>
<td>250</td>
<td>Address</td>
</tr>
<tr>
<td>homedir</td>
<td>Text</td>
<td>100</td>
<td>Home Directory</td>
</tr>
</tbody>
</table>

### 3.3.4 School

This table contains the information of School. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sch_id(PK)</td>
<td>Text</td>
<td>8</td>
<td>School ID</td>
</tr>
<tr>
<td>sch_name</td>
<td>Text</td>
<td>40</td>
<td>School Name</td>
</tr>
<tr>
<td>sch_desc</td>
<td>Text</td>
<td>250</td>
<td>School Description</td>
</tr>
</tbody>
</table>

### 3.3.5 Semester

This table contains the information of Semester. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sem_id(PK)</td>
<td>Text</td>
<td>8</td>
<td>Semester ID</td>
</tr>
<tr>
<td>beg_date</td>
<td>Date</td>
<td>-</td>
<td>Beginning Date</td>
</tr>
<tr>
<td>end_date</td>
<td>Date</td>
<td>-</td>
<td>Ending Date</td>
</tr>
</tbody>
</table>

### 3.3.6 Student

This table contains the information of Student. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>stu_id (PK)</td>
<td>Text</td>
<td>30</td>
<td>Student ID</td>
</tr>
<tr>
<td>first_name</td>
<td>Text</td>
<td>20</td>
<td>First Name</td>
</tr>
<tr>
<td>middle_initial</td>
<td>Text</td>
<td>20</td>
<td>Middle Initial</td>
</tr>
<tr>
<td>last_name</td>
<td>Text</td>
<td>20</td>
<td>Last Name</td>
</tr>
<tr>
<td>pass_wd</td>
<td>Text</td>
<td>20</td>
<td>Password</td>
</tr>
<tr>
<td>sch_id</td>
<td>Text</td>
<td>8</td>
<td>School ID</td>
</tr>
<tr>
<td>email_id</td>
<td>Text</td>
<td>40</td>
<td>Email ID</td>
</tr>
<tr>
<td>phone_no</td>
<td>Text</td>
<td>12</td>
<td>Phone Number</td>
</tr>
<tr>
<td>cell_no</td>
<td>Text</td>
<td>20</td>
<td>Cell Number</td>
</tr>
<tr>
<td>address</td>
<td>Text</td>
<td>250</td>
<td>Address</td>
</tr>
<tr>
<td>work_address</td>
<td>Text</td>
<td>250</td>
<td>Work Address</td>
</tr>
</tbody>
</table>
3.3.7 Sysadmin

This table contains the information of System administrator. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sys_id(PK)</td>
<td>Text</td>
<td>20</td>
<td>System Administrator ID</td>
</tr>
<tr>
<td>pass_wd</td>
<td>Text</td>
<td>20</td>
<td>Password</td>
</tr>
</tbody>
</table>

3.3.8 Tecepchair

This table contains the information of TECP chair. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chair_id (PK)</td>
<td>Text</td>
<td>20</td>
<td>Chair ID</td>
</tr>
<tr>
<td>first_name</td>
<td>Text</td>
<td>20</td>
<td>First Name</td>
</tr>
<tr>
<td>middle_initial</td>
<td>Text</td>
<td>20</td>
<td>Middle Initial</td>
</tr>
<tr>
<td>last_name</td>
<td>Text</td>
<td>20</td>
<td>Last Name</td>
</tr>
<tr>
<td>pass_wd</td>
<td>Text</td>
<td>20</td>
<td>Password</td>
</tr>
<tr>
<td>email_id</td>
<td>Text</td>
<td>40</td>
<td>Email ID</td>
</tr>
<tr>
<td>phone_no</td>
<td>Text</td>
<td>12</td>
<td>Phone Number</td>
</tr>
<tr>
<td>fax_no</td>
<td>Text</td>
<td>12</td>
<td>Fax Number</td>
</tr>
<tr>
<td>address</td>
<td>Text</td>
<td>250</td>
<td>Address</td>
</tr>
</tbody>
</table>

3.3.9 crs_image

This table contains the information of course and images of the course. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>img_id(PK)</td>
<td>Text</td>
<td>40</td>
<td>Image ID</td>
</tr>
<tr>
<td>crs_id</td>
<td>Text</td>
<td>8</td>
<td>Course ID</td>
</tr>
</tbody>
</table>

3.3.10 crs_test

This table contains the information of tests in a course. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>test_id</td>
<td>Text</td>
<td>20</td>
<td>Test ID</td>
</tr>
<tr>
<td>crs_id</td>
<td>Text</td>
<td>8</td>
<td>Course ID</td>
</tr>
<tr>
<td>ins_id</td>
<td>Text</td>
<td>20</td>
<td>Instructor ID</td>
</tr>
<tr>
<td>sem_id</td>
<td>Text</td>
<td>8</td>
<td>Semester ID</td>
</tr>
</tbody>
</table>
percent_weight | Integer | 3   | Percentage Weight
max_score       | Integer | 4   | Maximum Score
test_beg_date   | Date    | -   | Test Beginning Date
test_end_date   | Date    | -   | Test End Date
URL             | Text    | 100 | URL of test
numquestions    | Integer | 4   | Number of Questions

3.3.11 crs_test_image

This table contains the information of images of a test of a course. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>image_id</td>
<td>Text</td>
<td>40</td>
<td>Image ID</td>
</tr>
<tr>
<td>crs_id</td>
<td>Text</td>
<td>8</td>
<td>Course ID</td>
</tr>
<tr>
<td>test_id</td>
<td>Text</td>
<td>20</td>
<td>Test ID</td>
</tr>
<tr>
<td>label</td>
<td>Text</td>
<td>6</td>
<td>Label</td>
</tr>
<tr>
<td>question_id</td>
<td>Text</td>
<td>8</td>
<td>Question ID</td>
</tr>
<tr>
<td>x_val</td>
<td>Integer</td>
<td>4</td>
<td>X Coordinate</td>
</tr>
<tr>
<td>y_val</td>
<td>Integer</td>
<td>4</td>
<td>Y Coordinate</td>
</tr>
<tr>
<td>min_val</td>
<td>Integer</td>
<td>5</td>
<td>Minimum Value</td>
</tr>
<tr>
<td>max_val</td>
<td>Integer</td>
<td>5</td>
<td>Maximum Value</td>
</tr>
</tbody>
</table>

3.3.12 crs_test_question

This table contains the information of questions of a test of a course. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>question_id</td>
<td>Text</td>
<td>4</td>
<td>Question ID</td>
</tr>
<tr>
<td>crs_id</td>
<td>Text</td>
<td>8</td>
<td>Course ID</td>
</tr>
<tr>
<td>test_id</td>
<td>Text</td>
<td>20</td>
<td>Test ID</td>
</tr>
<tr>
<td>ques_desc</td>
<td>Text</td>
<td>250</td>
<td>Question Description</td>
</tr>
<tr>
<td>var1</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>min1</td>
<td>Float</td>
<td>-</td>
<td>Minimum Value</td>
</tr>
<tr>
<td>max1</td>
<td>Float</td>
<td>-</td>
<td>Maximum Value</td>
</tr>
<tr>
<td>var2</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>min2</td>
<td>Float</td>
<td>-</td>
<td>Minimum Value</td>
</tr>
<tr>
<td>max2</td>
<td>Float</td>
<td>-</td>
<td>Maximum Value</td>
</tr>
<tr>
<td>var3</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>min3</td>
<td>Float</td>
<td>-</td>
<td>Minimum Value</td>
</tr>
<tr>
<td>max3</td>
<td>Float</td>
<td>-</td>
<td>Maximum Value</td>
</tr>
<tr>
<td>var4</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>min4</td>
<td>Float</td>
<td>-</td>
<td>Minimum Value</td>
</tr>
<tr>
<td>max4</td>
<td>Float</td>
<td>-</td>
<td>Maximum Value</td>
</tr>
<tr>
<td>var5</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
</tbody>
</table>
3.3.13 ins_profession

This table contains the information of instructor, courses he/she is offering, semester the course is being offered and the institution instructor belongs to. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ins_id</td>
<td>Text</td>
<td>20</td>
<td>Instructor ID</td>
</tr>
<tr>
<td>sch_id</td>
<td>Text</td>
<td>8</td>
<td>School ID</td>
</tr>
<tr>
<td>sem_id</td>
<td>Text</td>
<td>8</td>
<td>Semester ID</td>
</tr>
<tr>
<td>crs_id</td>
<td>Text</td>
<td>8</td>
<td>Course ID</td>
</tr>
</tbody>
</table>

3.3.14 stu_academics

This table contains the information of scores obtained and grade of a test of a student. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>stu_id</td>
<td>Text</td>
<td>30</td>
<td>Student ID</td>
</tr>
<tr>
<td>crs_id</td>
<td>Text</td>
<td>8</td>
<td>Course ID</td>
</tr>
<tr>
<td>Field Name</td>
<td>Type</td>
<td>Size</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>----------</td>
<td>------</td>
<td>-----------------</td>
</tr>
<tr>
<td>test_id</td>
<td>Text</td>
<td>20</td>
<td>Test ID</td>
</tr>
<tr>
<td>marks_obtained</td>
<td>Integer</td>
<td>4</td>
<td>Marks Obtained</td>
</tr>
<tr>
<td>max_score</td>
<td>Integer</td>
<td>4</td>
<td>Maximum Score</td>
</tr>
<tr>
<td>grade</td>
<td>Text</td>
<td>1</td>
<td>Grade</td>
</tr>
</tbody>
</table>

### 3.3.15 stu_crs_info

This table contains the information of final score and grade obtained on a course of a student. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>stu_id</td>
<td>Text</td>
<td>30</td>
<td>Student ID</td>
</tr>
<tr>
<td>crs_id</td>
<td>Text</td>
<td>8</td>
<td>Course ID</td>
</tr>
<tr>
<td>sem_id</td>
<td>Text</td>
<td>8</td>
<td>Semester ID</td>
</tr>
<tr>
<td>finalscore</td>
<td>Integer</td>
<td>3</td>
<td>Final Score</td>
</tr>
<tr>
<td>finalgrade</td>
<td>Text</td>
<td>1</td>
<td>Final Grade</td>
</tr>
</tbody>
</table>

### 3.3.16 stu_crs_test

This table contains the information of all questions of a test for all students. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>stu_id</td>
<td>Text</td>
<td>30</td>
<td>Student ID</td>
</tr>
<tr>
<td>crs_id</td>
<td>Text</td>
<td>8</td>
<td>Course ID</td>
</tr>
<tr>
<td>test_id</td>
<td>Text</td>
<td>20</td>
<td>Test ID</td>
</tr>
<tr>
<td>question_id</td>
<td>Text</td>
<td>4</td>
<td>Question ID</td>
</tr>
<tr>
<td>ques_desc</td>
<td>Text</td>
<td>250</td>
<td>Question Description</td>
</tr>
<tr>
<td>var1</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>val1</td>
<td>Float</td>
<td>-</td>
<td>Value</td>
</tr>
<tr>
<td>var2</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>val2</td>
<td>Float</td>
<td>-</td>
<td>Value</td>
</tr>
<tr>
<td>var3</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>val3</td>
<td>Float</td>
<td>-</td>
<td>Value</td>
</tr>
<tr>
<td>var4</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>val4</td>
<td>Float</td>
<td>-</td>
<td>Value</td>
</tr>
<tr>
<td>var5</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>val5</td>
<td>Float</td>
<td>-</td>
<td>Value</td>
</tr>
<tr>
<td>var6</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>val6</td>
<td>Float</td>
<td>-</td>
<td>Value</td>
</tr>
<tr>
<td>var7</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>val7</td>
<td>Float</td>
<td>-</td>
<td>Value</td>
</tr>
<tr>
<td>var8</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
<tr>
<td>val8</td>
<td>Float</td>
<td>-</td>
<td>Value</td>
</tr>
<tr>
<td>var9</td>
<td>Text</td>
<td>20</td>
<td>Variable</td>
</tr>
</tbody>
</table>
3.3.17 stu_knowledge

This table contains the information of proficiency of a student in basic topics before he/she has access to TECP. The fields of the table are given below:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Type</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>stu_id</td>
<td>Text</td>
<td>30</td>
<td>Student ID</td>
</tr>
<tr>
<td>topic</td>
<td>Text</td>
<td>80</td>
<td>Topic</td>
</tr>
<tr>
<td>proficiency</td>
<td>Text</td>
<td>20</td>
<td>Proficiency</td>
</tr>
</tbody>
</table>
3.4 Entity Relation Diagram

The relationships among various entities in database can be shown through an Entity-Relationship (ER) diagram.

Figure 3.2 is the ER diagram of the OGS application

![ER Diagram]

Figure 3.2 ER diagram of application
3.5 Input/Output and Processing

The interface is through a set of HTML pages. All the input collected from the user and the processed output being sent to the user is done through the HTML or PDF pages.

The intermediate processing of data is being done by PHP and LaTeX. The PHP acts as an engine between HTML pages and the MySQL Database (see Figure 3.3). The data collected from the user through the HTML pages are processed by PHP and if required are sent to MySQL database where the database engine executes appropriate query and returns the value to PHP, which displays the processed data to the user in the form of HTML pages.

![Diagram of data flow](image)

Figure 3.3 Diagrammatic view of data flow
3.6 Protocols and Interfaces

The application runs on HTTP (Hyper Text Transfer Protocol). HTTP protocol is the most common and the most efficient protocol being used for all Web based applications. The input from the user/client (request) would go to the server through the HTTP protocol. Input, so obtained, from the user/client is processed by the server internally and sent back to the requested client through the HTTP protocol. An HTML page received by the user end acts as a client software/interface through which the user/client can send information to the Server. A PHP script at the server end works on the requested information, processes it as required, and then sends it back to the requested user in the requested format. This Client/Server communication runs purely on HTTP Protocol. Figure 3.4 shows the pictorial representation of the interaction between client and server using HTTP protocol.

![Diagram of protocols and interfaces]

Figure 3.4 Schematic diagram of protocols and interfaces
3.7 File Formats

The application has three kinds of data in it:

1) Text Data - used for all the data transfer between the user and the application,
2) Database - to store the student data and the course data in the database,
3) Images - images for some questions and the graphs.
4) PDF - downloadable PDF files for the students.

The text data are the data that the user keys in on being prompted. The data are his/her user ID, password, comments, keys to the questions, etc. The application too changes the data from the database into plain text data for the user to view.

The database is the text data that is obtained from the user and is thought to be relevant. Though majority of this data is textual it is stored in a relational database (MySQL) for all further querying and retrieval [Date 1995].

The images are the images some of the questions have in them. The graphs obtained from the analysis of data from the database, such as the overall performance of the class and related queries are in the form of images. These images generally are in PNG, JPEG, or GIF format.

The final quiz or test a student has to take would be in two formats:

1) HTML or online format and
2) PDF or printable format.

All the questions, images and the required choices are embedded in a PDF file and would be downloadable.
3.8 Data Dictionary

A data dictionary is a collection of descriptions of the data objects or items in a data model for the benefit of programmers and others who need to refer to them [Date 1995]. The following is the data dictionary for the OGS application. This data dictionary gives the description of the proposed data types for the OGS system (see Figure 3.5).

<table>
<thead>
<tr>
<th>Data field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stu_id</td>
<td>Unique ID given to a student.</td>
</tr>
<tr>
<td>Ins_id</td>
<td>Unique ID given to an Instructor.</td>
</tr>
<tr>
<td>First_name</td>
<td>First name of Instructor/Student.</td>
</tr>
<tr>
<td>Last_name</td>
<td>Last name of Instructor/Student.</td>
</tr>
<tr>
<td>Password</td>
<td>Password of Instructor/Student.</td>
</tr>
<tr>
<td>Phone</td>
<td>Phone number of Instructor/Student.</td>
</tr>
<tr>
<td>Email_id</td>
<td>Email ID of Instructor/Student.</td>
</tr>
<tr>
<td>Test_id</td>
<td>Unique Test ID given to every test conducted.</td>
</tr>
<tr>
<td>Crs_id</td>
<td>Unique ID to given every course offered.</td>
</tr>
<tr>
<td>Question_bank</td>
<td>Set of questions for every course.</td>
</tr>
<tr>
<td>Solution_set</td>
<td>Set of solutions for every course.</td>
</tr>
<tr>
<td>Beg_date</td>
<td>Beginning date for a test.</td>
</tr>
<tr>
<td>End_date</td>
<td>End date for a test.</td>
</tr>
<tr>
<td>Crs_details</td>
<td>Details of the course.</td>
</tr>
<tr>
<td>Sem_offered</td>
<td>Semester a course is being offered.</td>
</tr>
<tr>
<td>Sem_id</td>
<td>Unique ID given to the semester.</td>
</tr>
<tr>
<td>Question_id</td>
<td>Unique ID given to each question of a course.</td>
</tr>
<tr>
<td>Question</td>
<td>The question.</td>
</tr>
<tr>
<td>Solution</td>
<td>The solution.</td>
</tr>
<tr>
<td>Image</td>
<td>Image associated either with the question or the graph generated in the analysis of data.</td>
</tr>
<tr>
<td>Grade</td>
<td>Grade a student got in a test in a course.</td>
</tr>
</tbody>
</table>

Figure 3.5 Data Dictionary for OGS
4. IMPLEMENTATION

This section covers the implementation part of the OGS. A few features in the implementation of the OGS are described. These are:

- Dynamic generation of values for variables in a question
- Dynamic generation of the solution for each question
- Dynamic generation of values on images

4.1 Dynamic Generation of Values for Variables in a Question

The instructor creates questions for a test he/she is planning to offer. Each question can have a maximum of nine variables in it. Each variable is associated with a set of minimum and maximum values. All the variables, their corresponding minimum and maximum values are stored in a question databank. Each question is associated with nine answer choices too. These answer choices too are stored in the same question databank. Every test has an associated beginning date and an end date to it. When a student logs in into the OGS application, a script checks for the current date and looks if this date falls in between the beginning and end date of any test the students is registered for. If the current date falls between the beginning date and the end date of the test the student is prompted to take the test. All the dynamic variable of each question of the test are generated as soon as the student opts to take the test. These values are then stored in the database for the future reference. Each question has a formula associated with it, which is then extracted from the database. The extracted formula has the same set of variables as the question. The dynamically generated values of the variables are then substituted in the formula, the formula is computed, and the right solution is stored in the database. The
solution that the student keys in is compared with the right solution and then the student is graded basing on the number of attempts to solve the question.

4.2 Dynamic Generation of Solution for Each Question

As all the questions have dynamic variables in them it is not possible for the instructor to come up with the right solution for each question during its design. To solve this problem, the instructor is asked to key in upto a maximum of ten formulas one of which has to be the right formula. All the formulas from the question bank are extracted and then the dynamic values for the variables are substituted in all the formulas to obtain the answer choices. The answer choices are then stored in the PDF file and in the database for the student. When the student takes the test, the solution keyed in by the student is compared with the right solution and depending on the number of attempts, the student is given a score and a grade.

4.3 Dynamic Generation of Values on Images

Each question can be associated with an image. The instructor can specify the coordinates where he/she wants the value associated with the variable at the place to show up on the image. The GD (Graphic Devices) Lib of the PHP does this for the OGS. Any image associated with the question will also be generated at the same time with the values printed on the image. The image will be given a unique ID associated with the student, course and test. The question substituted with dynamic variables, and the images developed are stored in a database for the student to take test. All the questions and images are then embedded in a PDF file which is associated with the student ID, course
ID and test ID. The student can then either view the PDF file or take the test with the appropriate values embedded on the images.
5. TESTING AND EVALUATION

Attempts have been made to provide testing mechanisms for Web applications. In the recent past these mostly relied on a fragile manner of testing link validity and HTML syntax. These methods are only testing the interface, not the software. Better methods have been proposed which emphasize testing each component [Ghosh 1998]. Unit testing to test the output of each component is widely used along with common programming languages, ASP, JSP, Perl, PHP, and Cold Fusion. Although a vital step, this on its own does not provide the robustness needed. However, the logic of the entire working of a complex system may contain flaws that cannot be noticed by testing the functionality of each individual component [Koenig 1988].

Testing the Online Grading System has been done with three goals in mind:

- Testing for downloading speed
- Testing for security of data
- Testing for session consistency

5.1 Testing for Downloading Speed

The application uses a few images, which makes the application easy to download. Every test may have an average of four or five images (related to Physics), which are not too large in size. None of the pages of the application took more than two seconds to load on a PC with 500 MHz processor and 133 MHz bus speed.
5.2 Testing for Security of Data

The application is deployed on a Dell Server operating on Redhat Linux 8.0. Linux has the same security levels as UNIX. Both Linux and UNIX are considered to be very strong operating systems. The security of the application totally depends on the security features of Linux. The application has been tested by trying to insert data, retrieve data or perform any other operation on the data without logging into the application. Every time an operation was tried to perform the application would take the user to the login page and prompt the user to login.

5.3 Testing for Session Consistency

Session consistency is a frequent problem encountered in many Web based applications. The scripting software – PHP has advanced and secure API’s to deal with the session consistency. All the pages that access the database have the session feature enabled, thus making sure that the valid logged in user is accessing the database. A user with a valid user id and a password can start a session immediately after logging into the application, and the session ends only when the user logs out of the application.

In addition to the above mentioned tests the following tests have been conducted:

- Unit testing
- Acceptance testing
- Stress testing
- Regression testing

5.3.1 Unit testing
There are five modules in the OGS, all the modules have been tested for Input/Output. Required fields cannot be left empty. Server side scripting language (PHP) tests for the required fields and does database processes.

5.3.2 Acceptance testing

Acceptance testing makes sure that the results obtained after the required input is given conform to what is expected by the user and the databases. The application has been tested for acceptance testing by observing that the results match the output expected.

5.3.3 Stress testing

Sessions support through PHP and locking capability of MySQL on the databases make sure that synchronization and database inconsistency problems are avoided. No concurrency or synchronization problems have been found in the application. Application has been tested by five to seven users accessing the database simultaneously. All the processor intensive jobs and database intensive requests have been found to be working well.

5.3.4 Regression testing

All the above tests have been conducted on the application after the initial phase of testing to make sure that the application conforms to the levels of consistency and efficiency as expected by the user.
6. FUTURE WORK

A more efficient JavaScript interface can be added to the application to lessen the number of clicks a user needs to access the features of the application. As of PHP version 4.2.2 very less support has been extended towards embedding JavaScript code.

Graphical representation of the scores can be added to the application to make it easier for instructor to evaluate the performance of the students.

Categorizing the questions on difficulty level basis to automatically tailor the quizzes according to the student response can be added to make the quiz taking more challenging and efficient at analyzing the students’ knowledge over the subject.
7. RESULTS AND CONCLUSION

The Online Grading System has some advanced features that many other similar applications do not possess. These are dynamic image manipulation, ability to change the variables in a question randomly for different users and substitute them in the question, and the creation of downloadable PDF files for students are among the new features. Though much effort has been made to make the application bug free, many new features can be added after the current application is tested for consistency. Many of the problems that may crop up after the application has been deployed for a few months or more have not been checked due to infeasibility.

Better ways to make sure that the system would not crash are regular disk backups and regular database backup. More features for instructors can be added to the application to make it easier for instructors to use.


---

^1 This book has been consulted for over all guidance during the design process, so no specific reference quoted.
9. APPENDICES

Appendix A. Database Table Definition

Primary Tables

# Table structure for table 'Coordinator'
# Creation: Jul 15, 2003 at 07:20 AM
# Last update: Jan 28, 2004 at 02:39 PM
CREATE TABLE 'Coordinator' ( 
  'crd_id' varchar(20) NOT NULL default '', 
  'first_name' varchar(20) NOT NULL default '', 
  'middle_initial' varchar(20) NOT NULL default '', 
  'last_name' varchar(20) NOT NULL default '', 
  'pass_wd' varchar(20) NOT NULL default '', 
  'sch_id' varchar(8) NOT NULL default '', 
  'email_id' varchar(40) NOT NULL default '', 
  'phone_no' varchar(12) NOT NULL default '', 
  'fax_no' varchar(12) NOT NULL default '', 
  'address' varchar(250) NOT NULL default '', 
  PRIMARY KEY ('crd_id')
) TYPE=MyISAM COMMENT='Stores Local coordinator information.';

# Table structure for table 'Course'
# Creation: Oct 28, 2003 at 12:46 AM
# Last update: Apr 14, 2004 at 01:56 AM
CREATE TABLE 'Course' ( 
  'crs_id' varchar(8) NOT NULL default '', 
  'crs_name' varchar(40) NOT NULL default '', 
  'crs_dese' varchar(250) NOT NULL default '', 
  'crs_dir' varchar(40) NOT NULL default '', 
  PRIMARY KEY ('crs_id')
) TYPE=MyISAM COMMENT='Stores Course Information.';

# Table structure for table 'Instructor'
# Creation: Oct 25, 2003 at 11:16 PM
# Last update: Mar 03, 2004 at 10:08 AM
CREATE TABLE 'Instructor' ( 
  'ins_id' varchar(20) NOT NULL default '', 
  'first_name' varchar(20) NOT NULL default '', 
  'middle_initial' varchar(20) default NULL, 
  'last_name' varchar(20) NOT NULL default '', 
  'pass_wd' varchar(20) NOT NULL default '', 
  ...
'sch_id' varchar(8) NOT NULL default '',
'email_id' varchar(40) NOT NULL default '',
'phone_no' varchar(12) NOT NULL default '',
'fak_no' varchar(12) NOT NULL default '',
'address' varchar(250) NOT NULL default '',
'homedir' varchar(100) NOT NULL default '',
PRIMARY KEY ('ins_id')
) TYPE=MyISAM COMMENT='Stores the personal information of the Instructor.';

# Table structure for table 'School'
# Creation: Jun 09, 2003 at 03:31 PM
# Last update: Jun 09, 2003 at 03:31 PM
CREATE TABLE 'School' (
  'sch_id' varchar(8) NOT NULL default '',
  'sch_name' varchar(40) NOT NULL default '',
  'sch_desc' varchar(250) NOT NULL default '',
  PRIMARY KEY ('sch_id')
) TYPE=MyISAM COMMENT='Stores School Information';

# Table structure for table 'Semester'
# Creation: Jun 29, 2003 at 01:16 AM
# Last update: Jul 15, 2003 at 12:56 AM
CREATE TABLE 'Semester' (
  'sem_id' varchar(8) NOT NULL default '',
  'beg_date' date NOT NULL default '0000-00-00',
  'end_date' date NOT NULL default '0000-00-00',
  PRIMARY KEY ('sem_id')
) TYPE=MyISAM COMMENT='Stores Semester information.';

# Table structure for table 'Student'
# Creation: Sep 23, 2003 at 09:37 PM
# Last update: Sep 23, 2003 at 09:37 PM
CREATE TABLE 'Student' (
  'stu_id' varchar(30) NOT NULL default '',
  'first_name' varchar(20) NOT NULL default '',
  'middle_initial' varchar(20) default NULL,
  'last_name' varchar(20) NOT NULL default '',
  'pass_wd' varchar(20) NOT NULL default '',
  'sch_id' varchar(8) NOT NULL default '',
  'email_id' varchar(40) NOT NULL default '',
  'phone_no' varchar(20) default NULL,
  'cell_no' varchar(20) NOT NULL default '',

'address' varchar(250) NOT NULL default '',
'work_address' varchar(250) NOT NULL default '',
PRIMARY KEY ('stu_id')
) TYPE=MyISAM COMMENT='Stores personal information of the student.';

# Table structure for table 'Sysadmin'
# Creation: Jul 02, 2003 at 12:07 AM
# Last update: Jul 02, 2003 at 12:09 AM
CREATE TABLE 'Sysadmin' (  
'sys_id' varchar(20) NOT NULL default '',
'pass_wd' varchar(20) NOT NULL default '',
PRIMARY KEY ('sys_id')
) TYPE=MyISAM COMMENT='Stores Sysadmin User ID and Password.';

# Table structure for table 'tecpchair'
# Creation: Jul 17, 2003 at 10:06 AM
# Last update: Jan 28, 2004 at 01:22 PM
CREATE TABLE 'tecpchair' (  
'chair_id' varchar(20) NOT NULL default '',
'first_name' varchar(20) NOT NULL default '',
'middle_initial' varchar(20) NOT NULL default '',
'last_name' varchar(20) NOT NULL default '',
'pass_wd' varchar(20) NOT NULL default '',
'email_id' varchar(40) NOT NULL default '',
'phone_no' varchar(12) NOT NULL default '',
'fax_no' varchar(12) NOT NULL default '',
'address' varchar(250) NOT NULL default '',
PRIMARY KEY ('chair_id')
) TYPE=MyISAM;
Join tables

# Table structure for table 'crs_image'
# Creation: Nov 09, 2003 at 09:06 PM
# Last update: Apr 06, 2004 at 12:16 PM
CREATE TABLE 'crs_image' (  
  'crs_id' varchar(8) NOT NULL default '',  
  'img_id' varchar(40) NOT NULL default '',  
  PRIMARY KEY ('img_id')  
) TYPE=MyISAM COMMENT='Stores images of a course.';

# Table structure for table 'crs_test'
# Creation: Apr 14, 2004 at 01:21 AM
# Last update: Apr 14, 2004 at 02:33 AM
CREATE TABLE 'crs_test' (  
  'test_id' varchar(20) NOT NULL default '',  
  'crs_id' varchar(8) NOT NULL default '',  
  'ins_id' varchar(20) NOT NULL default '',  
  'sem_id' varchar(8) NOT NULL default '',  
  'percent_weight' int(3) NOT NULL default '0',  
  'max_score' int(4) NOT NULL default '0',  
  'test_beg_date' date NOT NULL default '0000-00-00',  
  'test_end_date' date NOT NULL default '0000-00-00',  
  'URL' varchar(100) NOT NULL default '',  
  'numquestions' tinyint(4) NOT NULL default '0'  
) TYPE=MyISAM COMMENT='Stores the test details of courses.';

# Table structure for table 'crs_test_image'
# Creation: Apr 05, 2004 at 06:58 AM
# Last update: Apr 14, 2004 at 07:55 AM
CREATE TABLE 'crs_test_image' (  
  'image_id' varchar(40) NOT NULL default '',  
  'crs_id' varchar(8) NOT NULL default '',  
  'test_id' varchar(20) NOT NULL default '',  
  'label' varchar(6) default NULL,  
  'question_id' varchar(8) default NULL,  
  'x_val' int(4) default '0',  
  'y_val' int(4) default '0',  
  'min_val' int(5) default '0',  
  'max_val' int(5) default '0',  
  FULLTEXT KEY 'crs_id' ('crs_id')  
) TYPE=MyISAM COMMENT='Stores information about an image parameters for a test.';
CREATE TABLE 'crs_test_question' (  
  'question_id' varchar(4) NOT NULL default '',  
  'crs_id' varchar(8) NOT NULL default '',  
  'test_id' varchar(20) NOT NULL default '',  
  'ques_dese' varchar(250) NOT NULL default '',  
  'var1' varchar(20) NOT NULL default '',  
  'min1' float NOT NULL default '0',  
  'max1' float NOT NULL default '0',  
  'var2' varchar(20) NOT NULL default '',  
  'min2' float NOT NULL default '0',  
  'max2' float NOT NULL default '0',  
  'var3' varchar(20) NOT NULL default '',  
  'min3' float NOT NULL default '0',  
  'max3' float NOT NULL default '0',  
  'var4' varchar(20) NOT NULL default '',  
  'min4' float NOT NULL default '0',  
  'max4' float NOT NULL default '0',  
  'var5' varchar(20) NOT NULL default '',  
  'min5' float NOT NULL default '0',  
  'max5' float NOT NULL default '0',  
  'var6' varchar(20) NOT NULL default '',  
  'min6' float NOT NULL default '0',  
  'max6' float NOT NULL default '0',  
  'var7' varchar(20) NOT NULL default '',  
  'min7' float NOT NULL default '0',  
  'max7' float NOT NULL default '0',  
  'var8' varchar(20) NOT NULL default '',  
  'min8' float NOT NULL default '0',  
  'max8' float NOT NULL default '0',  
  'var9' varchar(20) NOT NULL default '',  
  'min9' float NOT NULL default '0',  
  'max9' float NOT NULL default '0',  
  'solution1' varchar(100) NOT NULL default '',  
  'solution2' varchar(100) NOT NULL default '',  
  'solution3' varchar(100) NOT NULL default '',  
  'solution4' varchar(100) NOT NULL default '',  
  'solution5' varchar(100) NOT NULL default '',  
  'right_solution' varchar(100) NOT NULL default '',  
  'formula' varchar(250) NOT NULL default '',  
  'max_points' varchar(40) NOT NULL default '',  
  'neg_points' char(3) NOT NULL default ''
)
'max_time' char(3) default NULL,
PRIMARY KEY ('question_id')
) TYPE=MyISAM COMMENT='Stores the details of each question of a test';

# Table structure for table 'ins_profession'
# Creation: Jun 29, 2003 at 02:46 AM
# Last update: Jan 31, 2004 at 01:51 PM
CREATE TABLE 'ins_profession' (  
  'ins_id' varchar(20) NOT NULL default '',  
  'sch_id' varchar(8) NOT NULL default '',  
  'sem_id' varchar(8) NOT NULL default '',  
  'crs_id' varchar(8) NOT NULL default ''
) TYPE=MyISAM COMMENT='Stores the details of the courses offered by instructors in ';

# Table structure for table 'sch_crs'
# Creation: Jun 09, 2003 at 03:42 PM
# Last update: Jun 09, 2003 at 03:42 PM
CREATE TABLE 'sch_crs' (  
  'sch_id' varchar(8) NOT NULL default '',  
  'crs_id' varchar(8) NOT NULL default '',  
  'crs_desc' varchar(250) NOT NULL default ''
) TYPE=MyISAM COMMENT='Stores details of the courses offered at all the Universities';

# Table structure for table 'stu_academics1'
# Creation: Jan 06, 2004 at 01:32 PM
# Last update: Mar 03, 2004 at 10:09 AM
CREATE TABLE 'stu_academics1' (  
  'stu_id' varchar(30) NOT NULL default '',  
  'crs_id' varchar(8) NOT NULL default '',  
  'test_id' varchar(20) NOT NULL default '',  
  'marks_obtained' int(4) NOT NULL default '-999',  
  'max_score' int(4) NOT NULL default '0',  
  'grade' char(1) NOT NULL default ''
) TYPE=MyISAM COMMENT='Stores academic information related to student';
# Table structure for table 'stu_crs_info'
# Creation: Apr 05, 2004 at 04:46 PM
# Last update: Apr 05, 2004 at 04:46 PM
CREATE TABLE 'stu_crs_info' (  
    'stu_id' varchar(30) NOT NULL default '',  
    'crs_id' varchar(8) NOT NULL default '',  
    'sem_id' varchar(8) NOT NULL default '',  
    'finalscore' int(3) NOT NULL default '0',  
    'finalgrade' char(1) NOT NULL default '',  
    PRIMARY KEY ('crs_id','stu_id')  
) TYPE=MyISAM COMMENT='Store the final scores and grades if the students';

# Table structure for table 'stu_crs_test'
# Creation: Apr 14, 2004 at 08:04 AM
# Last update: Apr 16, 2004 at 01:23 PM
CREATE TABLE 'stu_crs_test' (  
    'stu_id' varchar(30) NOT NULL default '',  
    'crs_id' varchar(8) NOT NULL default '',  
    'test_id' varchar(20) NOT NULL default '',  
    'question_id' varchar(4) NOT NULL default '',  
    'var1' varchar(20) default NULL,  
    'val1' float default NULL,  
    'var2' varchar(20) default NULL,  
    'val2' float default NULL,  
    'var3' varchar(20) default NULL,  
    'val3' float default NULL,  
    'var4' varchar(20) default NULL,  
    'val4' float default NULL,  
    'var5' varchar(20) default NULL,  
    'val5' float default NULL,  
    'var6' varchar(20) default NULL,  
    'val6' float default NULL,  
    'var7' varchar(20) default NULL,  
    'val7' float default NULL,  
    'var8' varchar(20) default NULL,  
    'val8' float default NULL,  
    'var9' varchar(20) default NULL,  
    'val9' float default NULL,  
    'stu_response' varchar(80) default NULL,  
    PRIMARY KEY ('stu_id','crs_id','test_id','question_id')  
) TYPE=MyISAM;
# Table structure for table 'stu_knowledge'
# Creation: Sep 25, 2003 at 04:18 PM
# Last update: Sep 25, 2003 at 04:18 PM
CREATE TABLE 'stu_knowledge' (  'stu_id' varchar(30) NOT NULL default '',  'topic' varchar(80) NOT NULL default '',  'proficiency' varchar(20) NOT NULL default '',  PRIMARY KEY ('stu_id')  ) TYPE=MyISAM COMMENT='Stores the proficiency of the students in basic courses';
Appendix B. Source Code CD-ROM