GRADUATE STUDENT TRACKING SYSTEM

GRADUATE PROJECT

Submitted to the Faculty of
the Department of Computing and Mathematical Sciences
Texas A&M University-Corpus Christi
Corpus Christi, Texas

In Partial Fulfillment of the Requirements for the Degree of
Master of Science in Computer Science

by

Sonali Deshpande
Fall 2004

Committee Members

Dr. John Fernandez
Committee Chairperson

Dr. Chandrika Rao
Committee Member

Dr. David Thomas
Committee Member
GRADUATE STUDENT TRACKING SYSTEM

GRADUATE PROJECT

Submitted to the Faculty of
the Department of Computing and Mathematical Sciences
Texas A&M University-Corpus Christi
Corpus Christi, Texas

In Partial Fulfillment of the Requirements for the Degree of
Master of Science in Computer Science

by

Sonali Deshpande
Fall 2004

Committee Members

Dr. John Fernandez
Committee Chairperson

Dr. Chandrika Rao
Committee Member

Dr. David Thomas
Committee Member
ABSTRACT

In this project a Graduate Student Tracking System (GSTS) is designed in order to track students from the start to finish of their program. This system is mostly useful to the graduate coordinators to maintain and track each student’s information. Currently, many graduate coordinators have to go through the University’s admission and records system to procure information on any student. These admission and records systems are centrally managed by the administrators and do not allow the coordinators to make changes. The GSTS allows the graduate coordinators to control the student’s record on their own computer without having to go through the University’s system. Among its many functions, the GSTS allows coordinators to pull a student’s academic record, give “clearance” to allow students to register for a course provided the pre-requisite requirements are met, and to print various reports in order to study academic performance.

Students also benefit because of the GSTS. The students can check but not change their grades and can upgrade personal data. GSTS not only lets the students print their timetables, but also lets them view their entire academic record and register or drop the courses online. The system also keeps track of all the pre-requisites and graduate level courses completed by the student.
# TABLE OF CONTENTS

Abstract ................................................................................................................ ii

Table of Contents ................................................................................................ iii

List of Figures ....................................................................................................... vi

1. Background and Rational .............................................................................. 1

2. Graduate Student Tracking System .......................................................... 6

   2.1 Overview .............................................................................................. 6

   2.2 Home Page .......................................................................................... 8

   2.3 Login Page ........................................................................................... 8

   2.4 Registration Page ............................................................................... 9

   2.5 Student Main Menu .......................................................................... 11

   2.6 Change Password ............................................................................. 17

   2.7 Forgot Password .............................................................................. 18

   2.8 Administration Page ......................................................................... 19

   2.9 Add Student ...................................................................................... 19

   2.10 Find Student .................................................................................. 21

   2.11 Administration Main Menu ........................................................... 22

   2.12 Enter Grades .................................................................................. 25

   2.13 Graduate Catalogue ......................................................................... 27

   2.14 Class Schedule ............................................................................. 29

   2.15 Print Reports .................................................................................. 33

3. System Design ............................................................................................... 35
3.1 Environment...........................................................................35

3.1.1 System Components..............................................35

3.1.2 Programming Languages .........................36

3.1.3 Security......................................................37

3.2 Database Design......................................................38

3.3 Major Page Design.....................................................44

3.3.1 Login Page ......................................................44

3.3.2 Registration into GSTS.................................44

3.3.3 Student Main Menu.................................44

3.3.4 Student Degree Plan.................................45

3.3.5 Course Grades...............................................45

3.3.6 Course Catalogue........................................45

3.3.7 Class Registration..........................................45

3.3.8 Drop Course...................................................45

3.3.9 Administration Main Menu...........................46

3.3.10 Edit Grades.....................................................46

3.3.11 Add Course....................................................46

3.3.12 Add Student....................................................46

3.3.13 Enter Grades....................................................47

3.3.14 Graduate Catalogue.................................47

3.3.15 Print Reports....................................................47

4. Implementation .................................................................48
LIST OF FIGURES

Figure 2.1.  Structure Chart.................................................................7
Figure 2.2.  Home Page.................................................................8
Figure 2.3.  Login Page.................................................................9
Figure 2.4a.  Not A Member..............................................................9
Figure 2.4b Registration Form.........................................................10
Figure 2.4c.  Registration Error Message ........................................11
Figure 2.4d.  Registration Successful Message ................................11
Figure 2.5a.  Student Main Menu ..................................................11
Figure 2.5b.  Student Degree Plan ..................................................13
Figure 2.5c.  Select Semester Screen .............................................14
Figure 2.5d.  Course Grades ............................................................14
Figure 2.5e.  Transcripts ................................................................14
Figure 2.5f.  Business Class Schedule ...........................................15
Figure 2.5g.  Class Registration Screen ........................................15
Figure 2.5h.  Class Registration Output .........................................16
Figure 2.6a.  Change Password Screen .........................................17
Figure 2.6b.  Password Changed Message .....................................17
Figure 2.7a.  Forgot Password Form ...............................................18
Figure 2.7b.  Reset Password ..........................................................18
Figure 2.8.  Administration Page ....................................................19
Figure 2.9  Add Student Form ........................................................20
Figure 3.1  Data Flow of the System’s Components .......................................36
Figure 3.2  ER Diagram of the GSTS ..............................................................41
1. BACKGROUND AND RATIONALE

High quality information is a prerequisite for effective decisions. A good information system allows university administrators to make better decisions [Holmes 1998]. Administrators play an important role in terms of managing a program as well as coordinating activities in the program [Holmes 1998]. Online systems are not only beneficial to administrators, but students also benefit from such systems, which allow them to manage their courses and degree plans [Thomas 1999]. Graduate coordinators are individuals who advise graduate students on a wide variety of issues, ranging from registration procedures to professional development. They often play a major role in the management of a department's admissions process and awarding of financial aid. They help international graduate students from all over the world adjust to the U.S. educational system. In certain institutions graduate coordinators are the sole decision makers with respect to graduate student admission.

Not all universities have the resources to buy expensive specialized, fault-tolerant computing systems. Some universities opt to develop their own registration system in-house. The University of Newcastle in the United Kingdom decided to implement such an in-house system to assist their students in registering for classes [Little 1999]. Their system, named Arjuna, implemented in C++, was based on the computational model of nested atomic actions controlling operations on persistent objects. “Atomic actions guarantee consistency in the presence of failures and concurrent users, and Arjuna objects can be replicated on distinct nodes in order to obtain high availability”.
The Graduate Student Tracking System (GSTS) is an interactive system designed
to assist graduate coordinators in maintaining graduate student information. This system
is designed specifically for those coordinators that are the sole decision makers with
respect to graduate admissions. Currently, graduate coordinators of Texas A & M
University – Kingsville, have to depend on their University’s Office of Admission and
Records (OAR) for information relating to their graduate students. But with the GSTS
the coordinators are able to track each student’s progress right from the application for
admission to the completion of their program. The system will also generate specific
reports to be sent to the registrar and other administrative officers in the university. The
students will be able to view their grades online but not change them. They will also be
able to print their schedules, student record and graduate catalogue. The graduate
coordinator will be able to add on, delete or modify a student’s record. He will be able to
electronically enter the grades for the courses that are completed by the student at the
university. Most importantly, the coordinator will be able to keep track of all the courses
the student has completed and all the courses that are yet to be completed. This way, the
student will not be able to register for a course if the pre-requisite requirements are not
met [Tran 2001].

One of the problems that graduate coordinators face is the speed at which the
OAR provides information necessary to make decisions regarding the graduate programs.
Since most graduate coordinators are professors who take on the additional responsibility
of being a coordinator, it becomes time consuming and tedious to frequently call the
OAR for information. Even if the OAR has computerized the admissions process,
graduate coordinators do not have access to change the information or to get specific reports that they may need in order to make college-level or departmental-level decisions.

The University of Michigan [Michigan 1997] developed a system, which allows a student to register for a schedule of classes for a term. In order to do so, the system retrieves data from a degree audit report, the class master schedule, and a student’s course selections. One of the unique features of the University of Michigan system is the fact that students are allowed to select preferred courses as well as preferred instructors, days, times and other features. The following steps portray the operations of the University of Michigan system:

1. Student accesses electronically the degree plan to determine the requirements s/he has remaining and the courses, which will satisfy those requirements.

2. The student will select the preferred courses and will indicate days and times, which are not available for scheduling. The student may also indicate preferred instructor, times, days, and other features.

3. The system will access the course master schedule to arrange a schedule of classes and will report the tentative results to the student.

4. The student may select a schedule of classes from the schedules produced by the system or may determine a schedule.

In order to accomplish its objectives, the University of Michigan system accesses the curriculum database to enforce all course restrictions and requirements. Students have access to information about a course or a section of a course, including restrictions, credits, instructor, time and day offered, requirements, and the course syllabus easily and readily. The system will also collect unmet student demand for the course if the student
tries to register for a course, which is closed. This system is also designed to request special permission to register for courses. All information relating to the special permission will be provided electronically. Finally, the system provides a complete statement of account at the conclusion of registration, including the tuition and fees for the registration session.

Another distinguishing feature of the University of Michigan system is the ability to collect requests for courses for future terms. Data from unmet student requests and from requests for future terms will be available to the departments for planning the course master schedule for future terms.

The Graduate Student Tracking System (GSTS) is designed to assist graduate coordinators in managing and tracking graduate student progress. The University of Newcastle system [Little 1999] as well as the University of Michigan system [Michigan 1997] were designed as a university-wide registration system. The scope of GSTS is limited to the College of Business Administration at the Texas A&M University – Kingsville given the cost and time constraints faced by the developer of the GSTS.

Both the University of Newcastle and the University of Michigan spent significant time, money, and human resources developing their respective systems. As a result their systems have more feature than the GSTS. For example, at the University of Michigan, department chairs can plan course schedules for future semesters based on the inputs provided by students in terms of requests for specific courses. Despite the fact that the University of Michigan system is a university-wide system, there are some similarities between their system and the GSTS. Students can register online on the GSTS similar to students registering on the University of Michigan or the University of
Newcastle systems. Administrators can also use the GSTS to generate several reports regarding student status, performance, and progress. Thus it is evident that although the GSTS is narrower in scope, it is in many ways similar, albeit a smaller version, of larger university-wide systems.
2. GRADUATE STUDENT TRACKING SYSTEM

2.1 Overview

In this project an online Graduate Student Tracking System (GSTS) is designed to help the graduate coordinators maintain and track graduate student information. This system functions through a Web site that is accessible through most of the Web browsers that support HTML and JavaScript. The design is based on three sub-systems. The login sub-system blocks unqualified access to other sub-systems. In the student sub-system, student users can see and print their student records and timetables, as well as change their passwords. They are able to register for courses online and drop them as well. They are also able to view their grades and get an unofficial copy of their transcripts. The administrative sub-system allows the administrative users to access these functions plus a superset of capabilities which includes adding new students, deleting students, changing the information recorded for a current student and also checking to see if a student has met the pre-requisite requirements in order to register for a graduate course. The administrative users are able to print various types of reports that will assist them in making department-level decisions.

The system is written for a multi-user environment and implemented with user-friendly features. This includes navigating with a mouse to select a button or link that initiates an action to lead to a new page or retrieve data from the database. The links and buttons direct users to view the desired properties, choose what page they would like to view, or choose the option to exit the site at any time [Lemay 1995]. Figure 2.1 shows the structure chart for the Graduate Student Tracking System.
2.2 Home Page

The **Home Page** (Fig 2.2) is the first page a user sees when the Web site is accessed. It gives a welcome message. The university can customize this page by placing the university’s logo above the welcome message. Clicking on the “**Services**” brings up the **Login Page**.

![Home Page](image)

**Figure 2.2 Home Page**

2.3 Login Page

This screen (Fig 2.3) requires keyboard input. Users see two boxes for entry of a user identification number and password. The user identification number is the social security number of the user. The purpose of this screen is to identify a user as a student or an administrator. If the user identification number given is that of an administrative user, the system displays the **Administrative Page**, otherwise the **Student Main Menu** is displayed. If the user inputs an invalid password and/or identification number, an error message and a prompt for re-entry is displayed.
2.4 Registration Page

When a user, who has not yet registered, clicks on the “Not a Member” link, a screen (Fig 2.4a) is displayed asking the user if he has been admitted into the graduate program. If the user selects the “No” button, he is taken back to the Login Page and if he selects the “Yes” button, then the Registration into the Graduate Student Tracking System page (Fig 2.4b) is displayed.
If a user, who has not yet been admitted into the graduate program, tries to register into the system, he will get an error message and will not be registered.

**Figure 2.4b Registration Form**

The registration into the GSTS page requires the user to fill out a form containing his information. The first field in the form is the User ID, where the user enters his social security number. This user ID and the password provided by the user will log the user into the system. If the user enters an incorrect user ID or if the password is less than 6 characters or greater than 16 characters, an error message is issued and the user is prompted to enter these fields again (Fig 2.4c). Similarly, this page also checks if the format of the phone number is correct and also validates the email address provided. The “Graduate Major” field is a drop box, which contains two options “Business” and “Accounting”. The graduate students have the option of majoring either in business or in accounting. Both business and accounting have different course catalogues and different pre-requisite requirements. Once all the information is verified the user is registered into the system (Fig 2.4d).
2.5 Student Main Menu

After the user enters a valid identification number and password, the system identifies the user as an administrator or a student. If the user is identified as a student, the Student Main Menu (Fig 2.5a) is displayed on the screen. It is an interface that allows a user quick and easy access to specific sections of the student system.
In the upper left corner of the screen, the student sees an area that contains his personal information. This information serves to confirm to the user that his information is correct and up to date. The student is not given access to change any of this information directly.

Under the student information is another sub-window containing the student’s personal timetable. This window displays the student’s current semester’s timetable. This timetable displays all of the courses that the student is currently registered in, their timing and the room number. The timetable is scrollable both horizontally and vertically.

In the upper right corner important dates are displayed. “Today’s Date”, “Last Day of Registration” and “Drop Date” serves to remind the student of registration deadlines. Students cannot change these dates.

Below these dates there are six buttons. When the “Degree Plan” button is selected the Student Degree Plan is displayed (Fig 2.5b). The two majors (Business and Accounting) have different degree plans, with different core courses, different electives and core electives. Hence when the “Degree Plan” button is selected, the system has to first check the major of the student and then display the major specific degree form. The degree form displays all the required pre-requisites, core courses, core electives and electives for that major. All the courses that have been waived or completed by the student are displayed as “completed” or “waived” in the textbox in front of the respective course and the textboxes in front of the courses that are empty, are the courses the student needs to complete in order to meet the degree requirements. This is a helpful feature, as the student is able to get all this information online and need not go to the Admissions office or to the coordinator to get the same. If a student completes the entire credit hours
requirement for the degree program, his record indicates it by displaying a message “All requirements for the degree program completed”.

Figure 2.5b Student Degree Plan

The student is also able to view his grades by clicking on the “Course Grades” button. On selecting this button, the user is asked to select a semester from a drop box (Fig 2.5c) and then the grades are displayed for the courses registered in that semester (Fig 2.5d). Similarly, the student can get an unofficial copy of the transcript by selecting the “Transcripts” button from the Student Menu (Fig 2.5e).
On clicking the "Class Schedule" button, the student is again asked to select a semester and a major (Business or Accounting) and then he is able to view all the courses offered in that semester along with their course number, time, days offered, location, pre-requisites required for a course and the instructor’s name (Fig 2.5f).
An important feature in the Student Menu is that the student is allowed to register for the classes online. On selecting the “Registration” Button, the student is first asked to select a semester for which he wants to register and then a form is displayed which allows the student to enter up to five class codes (Fig 2.5g).

After submitting this information, the system checks if the class codes entered are valid. It then checks if the course is offered in that semester. If not, an error message is
issued. Similarly, the system checks for various other requirements for example, if all the pre-requisites for the course are met. If the student does not complete any one of the pre-requisites, the system does not allow him to register for that course. If a particular course requires permission from the instructor to register, the system gives an appropriate message. The student has to meet with the instructor and if the instructor gives permission to register, only then can the student register for that course. Similarly, if the GPA is less than 3.0 or if his admission status is probationary, the student cannot register. The system also checks to see if there are any timing clashes for the courses the student is registering in. If any, the system gives an appropriate message. If the student enters a class code for the course he has already completed or has been waived for him, the system checks the database and gives an appropriate message and then prompts the user to enter another class code (Fig 2.5h). This system also allows the user to drop the courses online before the deadline, by selecting the “Drop Class” link from the Student Menu. After the deadline, if the student tries to drop a course, a message is issued notifying the student to go to the Admissions Office.

Figure 2.5h Class Registration Output
At the bottom of the screen, the user menu is displayed. The “Change Password” link lets the student change his password, the “Main Menu” link takes him back to the Student Main Menu, the “Home Page” link takes him back to the home page and the “Logout” link logs the student out of the system.

2.6 Change Password Screen

To change their password, the user is able to click “Change Password” button from the user menu, which is displayed at the bottom of every screen. This creates a “Change Password” window (Fig 2.6a) in the center of the screen. User is then prompted to enter their old password, a new password and to verify their new password. He must then click on “Change Password” for the system to accept the request. If the password change is successful a confirmation window (Fig 2.6b) relays that information to the user. The passwords are not echoed on the screen for security reasons.

![Figure 2.6a Change Password Screen](image1)

![Figure 2.6b Password Changed Screen](image2)

The password change may be unsuccessful because the old password the user typed in did not match the password on file or their new password did not verify. In either case, an error window informs the user about the error and they must click “Back” button to get to the “Change Password” window where they can try again.
2.7 Forgot Password

The user is allowed to reset the old password in case he forgets his password. A link “Forgot password” is provided on the login page. On selecting this link, the user is prompted to enter his user ID (Fig 2.7a). When the user enters the ID and selects the “Reset Password” button, the system resets the old password to a random number, selects the email address of the user from the database and issues a notification of the changed password via email to the user (Fig 2.7b).

**Figure 2.7a Forgot Password Form**

**Figure 2.7b Reset Password**
The user is then allowed to log in using this password and then he can change the password of his choice using the “Change Password” link.

2.8 Administration Page

If the system identifies the user as an administrator, then the Administration Page is displayed (Fig 2.8). The administrator can perform various functions like find a student using his identification number or his last name or print various kinds of reports or view online graduate catalogue or just add a new student to the database.

2.9 Add Students

When the administrator issues admission to the student for the graduate program, he is able to add a new student to the database by clicking on the “Add Student” button in the Administrative Page. On clicking this button, a form (Fig 2.9) is displayed. The administrator is able to enter the student’s information in this form. If the form is not filled out correctly, an error message is displayed on screen. The “International Student” field in the form is a drop box, which contains two options namely “YES” and
“NO”. If the student is a U.S citizen, the administrator enters “NO” in the “International Student” field, else he enters “YES”. If the field contains “YES”, the administrator has to enter the TOEFL score. If the TOEFL score is not entered, an error message is displayed and the administrator is prompted to enter the score. For a U.S citizen, the “TOEFL score” field remains blank.

![Figure 2.9 Add Student Form](image)

The administrator also has to enter the GMAT score for the student. If the administrator enters a GMAT score greater than 800, the system will give an error message. After all the information is entered, the administrator selects the “Register” button and the student record is saved in the database. Selecting the “Administration Page” link from the user menu at the bottom of the screen takes the user back to the Administration Page.

Only when the administrator enters the student in the database, can that student fill out the “Registration Form” and get logged into the system. If the student tries to register before the administrator issues him admission to the graduate program, the system gives an error message and does not log the student in.
2.10 Find Student

Administrators spend much of their time retrieving, validating and changing student information. Hence, find student function is a useful feature. Clicking on the “Find Student” button, gives the administrators two options for searching the student. The administrator can either search for a student using his User ID or his last name (Fig 2.10a). If the administrator opts for User ID, then a screen is popped requesting the student’s User ID (Fig 2.10b). The administrator then enters the number to retrieve the student’s record. It is possible to enter incorrect or nonexistent student numbers here. The system informs the user as to which type of error was made and return to the User ID prompt to allow the user to re-enter.

Similarly, if the administrator opts for the student’s last name, then a window is popped up requesting the student’s last name. Again, if the last name is misspelled or nonexistent an error message is displayed. It is possible that there may be more than one student with the same last name. In this case, the system displays the names and user ID’s of all the students with that last name (2.10c). When the administrator selects one of the
User ID’s, the Administration Main Menu page is displayed which contains the information of the student with that user ID.

![Administration Main Menu](image)

**Figure 2.10c Multiple Last Names**

### 2.11 Administration Main Menu

When the administrator enters the User ID or the last name of the student, the system retrieves the student’s information. The Administration Main Menu (Fig 2.11a) shows the name, ID number, address, undergraduate degree, major, admission status and the current semester of the student at the upper left corner.

Below the student’s information is the “Course Listing” sub-window. The window displays the courses the student is currently registered in, the courses he has completed or the courses that have been waived for him. This is very convenient for the administrator, as he is able to view the courses that are completed and waived for the student in the same window and does not have to go through different pages to view this information.
Next to the “Course Listing” sub-window is another sub-window that contains four buttons. If the administrator needs to change the grades of the student, he selects the “Edit Grades” button. On selecting this button, a screen is displayed which contains all the courses completed by the student in the current semester and their respective grades (Fig 2.11b). The administrator can change the grade of the student and selecting the “Save” button, will save the grade in the database.
The “Add Course” link on the Administration Main Menu lets the administrator enter the status for a course. For example, if the administrator waives a course for the student, he enters the course number and the status, which is “waived”. This information gets stored in the database after the administrator selects the “Submit” button. Similarly, if a student completes a course the administrator inputs “completed” in the status field. If a course requires permission from the instructor to register, the administrator inputs “permitted” in the status field after verifying if all the requirements for that course are met (Fig 2.11c).

![Add Course Screen](image)

Figure 2.11c Add Course Screen

If the course number entered is nonexistent, the system notifies the administrator and allows him to reenter (Fig 2.11d). If the administrator enters the course number and keeps the status field empty, the system prompts him to enter the status field.
Like the students, the administrator is also able to view the degree plan for the students by selecting the “Degree Plan” button from the Main Menu. This keeps the administrator current with the credit hours the students has completed towards the requirement of his degree. The administrator is able to see the same screen as displayed in Fig 2.5b. Similarly, the administrator is able to delete a student from the student or modify his information by selecting the “Delete Student” or “Modify Student” buttons respectively.

At the bottom of the screen, the user menu is displayed. The “Change Password” link lets the administrator change his password, the “Main Menu” link takes him back to the administrative page, the “Home Page” link takes him back to the home page and the “Logout” link logs the administrator out of the system.

2.12 Enter Grades

The “Enter Grades” button allows the administrator to enter the grades of the student for each course. On selecting this link, the administrator is prompted to enter the
current semester. After he enters the semester, a pop up window is displayed which consists of all the courses that are offered in that semester (Fig 2.12a).

Figure 2.12a Select Course Screen

When the administrator wants to enter the grades for a course, he selects the course from the drop box and a window is displayed consisting of all the names of the students currently registered for that course (Fig 2.12b). The administrator enters the course status (for example: “completed” or “incomplete”) and grades of each student and selects “Submit” to update the database with the information.

Figure 2.12b Enter Grades
2.13 Graduate Catalogue:

The administrator is able to view the graduate catalogue online for Business as well as Accounting. The Graduate Catalogue (Fig 2.13a) consists of list of all the courses numbers for Business and Accounting major along with the course name, prerequisites required for each course, a short description of the course and the semester in which they are offered. The administrator has the option of editing, deleting and inserting a course.

![Figure 2.13a Graduate Catalogue](image)

On selecting the “Delete” button, a screen is displayed (Fig 2.13b) asking the administrator if he wants to delete that course. If the administrator selects the “No” button, he is taken back to the Graduate Catalogue screen and if he selects the “Yes” button, the course is deleted from the Graduate Catalogue as well as from the Class Schedule. Similarly, selecting the “Edit” button will let the administrator make changes to the course (Fig 2.13c) and selecting the “Save” button will update the Graduate
Catalogue and Class Schedule with the changes. Upon successfully update of the changes into the database, a screen as shown in Fig 2.13d is displayed.

The administrator is also able to insert a new course into the Graduate Catalogue by selecting the “Insert Course in Catalogue” button. A screen will be displayed (Fig 2.13e) asking the administrator to fill out the information regarding the new course like the name of the course, the description of the course, the prerequisites required for the course, the semester the course is offered etc and on clicking the “Save”
button will store the information into the database. Upon successfully enter of the course into the database, a confirmation screen is displayed (Fig 2.13f), else an appropriate error message is given.

Figure 2.13e Insert Course into Catalogue

2.14 Class Schedule

The administrator is able to view the class schedule for both the majors by selecting the “Class Schedule” link from the menu in the Administration Page. The administrator is asked to select either the “Business Class Schedule” or the “Accounting Class Schedule” from the drop box and also a semester (Fig 2.14a). If the administrator selects the “Business Class Schedule” and “Fall” semester, the following page is displayed (Fig 2.14b). The students are able to see the same catalogue from the Student Page, except they are not allowed to change any field. The administrator has an option to add, drop and modify a course in the schedule.
If the administrator wants to edit a record, he is able to select the record and on clicking the “**Edit**” button, the following screen is displayed (Fig 2.14c). The administrator is able to edit the “**Time**”, “**Room No**”, “**Adm ID**” and “**Days**” the course is offered and on selecting the “**Save**” button, the information is saved in the database.
for any reason, the database cannot be updated, then an appropriate error message is displayed.

**Figure 2.14c Edit Course in Class Schedule**

The administrator also has the option of deleting a course from the Class Schedule for the selected semester and inserting another course. On selecting the course using the radio button and clicking on the “Delete” button will display the Delete a Course page (Fig 2.14d). Selecting the “Yes” button will delete the course from the Class Schedule, but not from the Graduate Catalogue. When the course is deleted successfully from the database, a confirmation message is displayed.
Similarly, the administrator will be able to insert a course into the Class Schedule by selecting the “Insert a Course” button (Fig 2.14e). The administrator will not be able to insert a new course, but a course, which is already present in the Graduate Catalogue.

If the course is inserted successfully into the class schedule, then the following message is displayed (Fig 2.14f), else an appropriate error message is displayed.
Figure 2.14f Confirmation message for inserting a course

2.15 Print Reports

The system allows the administrator to print various kinds of reports. Clicking on the “Print Report” button from the Administration Page, the administrator is able to see the following screen (Fig 2.15a). The screen has two drop-boxes. The first drop box has various options like “First Name”, “Last Name”, “GMAT Score”, “Current G.P.A” and so on. The second drop box has options as “Business”, “Accounting”, “Both”. Choosing any one of the options from both the drop boxes and clicking on the “Submit” button helps to sort the report by that field.
A sample report is shown in Fig 2.15b. Choosing the “Current GPA” field and the “Both” field from the second drop box creates this report. The GPA’s in the descending order sorts the report. The administrator is able to print the reports by clicking on the “Print Report” button.
3. SYSTEM DESIGN

3.1 Environment

This project is a cross-platform Web application. It can be accessed using any major browser that supports HTML 4.0 and JavaScript. It is implemented on Windows XP with Apache Server running. The MySQL relational database management system is used to store and manage data for the system. PHP is used to query and access the MySQL database. HTML and JavaScript are applied for generating the Web user interface. To run this project, a computer with a minimum of a 3.2 GB Hard Disk Drive and 32 MB RAM memory will be required.

3.1.1 System Components

As a Web application, the system includes three major components: Web browser, Web server, and database. The Web browser supports HTML 4 and JavaScript. With it, the user can request data and use the functions of the system. The Web server processes the user requests, sends data to the database or delivers the requested data back to the user. The system is implemented on Apache web Server or any other Web Server that supports PHP scripts. MySQL provides rich and useful functions. It includes [MA 2001]:

- Multi-threading. This means it can easily use multiple CPUs if available.
- A very fast thread-based memory allocation system and very fast joins using an optimized one-sweep multi-join.
- Functions that are implemented through a highly optimized class library. Usually there is not any memory allocation at all after query initialization.
- A privilege and password system that is very flexible and secured, and allows host-based verification. Passwords are secured because all password traffic is encrypted when one connects to a server.

These important characteristics make MySQL highly suited for accessing database on Internet. The system uses MySQL database to store and manage the data. The data flow and interaction between the system’s components are presented in the following figure.

Figure 3.1. Data Flow of the System’s Components

3.1.2 Programming Languages

In the project, HTML 4 is used to generate the static Web pages. JavaScript is chosen to create dynamic Web pages on the client side. JavaScript is very powerful in developing Web applications. It is a cross-platform scripting language supported by both Netscape and Internet Explorer.
PHP is used to generate server-side scripts to communicate between the Web server and the database. Like any other CGI program, PHP [Meloni 2000] can access databases and generate content on the fly, or create a Web interface for adding, deleting, and modifying elements within the database. It can work with just about any combination of Web server, operating system, and database one can think of. PHP user authentication can restrict access to the Web site. The goal of the language is to allow Web developers to write dynamically generated pages quickly.

3.1.3 Security

This Web-based tool uses session cookies to make the system more secure and allows a user to carry information across pages of the system. PHP session management addresses the problem of how to maintain user-state information in a stateless http environment. A cookie can be assigned for each page, but it won’t be a good solution when the users are browsing in a multi-user environment such as a computer lab or library, as not all users have cookies turned on. Sessions work by preserving the state of variables from page to page by tying them to a unique session ID on the server. PHP provides a session_start( ) function that creates a new session and subsequently identifies and establishes an existing one. Either way, a call to the session_start( ) function initializes a session. The first time a PHP script calls session_start( ), a session identifier is generated, and, by default, a Set-Cookie header field is included in the response. The response sets up a session cookie in the browser with the name PHPSESSID and the value of the session identifier. The session identifier (ID) is a random string of 32 hexadecimal digits, such as
A temporary file is created for the lifetime of the session, and all registered session variable information is written to that file.

When the user logs off, the sessions are destroyed. For example, when a user logs out of an application, a call to the `session_destroy()` function is made. A call to `session_destroy()` removes the session file from the system. After a PHP session is destroyed, the session cookie set on the user's computer is worthless, because it refers to a session that no longer exists. As well, the cookie contains no user information (user name, encrypted password, etc), only the worthless PHP session id.

While it is good practice to end a session with a call to `session_destroy()`, there is no guarantee that a user will log out by requesting the appropriate PHP script. PHP session management has a built-in garbage collection mechanism that ensures unused session files are cleaned up. This is important for two reasons: it prevents the directory from filling up with session files that can cause performance to degrade and, more importantly, it reduces the risk of someone guessing session IDs and hijacking an old unused session.

### 3.2 Database Design

The design of the database is based on the MySQL relational database. Figure 4.2 shows the ER diagram of the Graduate Student Tracking System. It is used to identify the data objects and their relationship using a graphical notation. It consists of the following tables:

- Student table
- Administration table
- Graduate Catalogue table
Student Table

Student table provides a location for storing the information about each student. This makes the adding, deleting and changing the student’s information easy and efficient. It contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stu_ID</td>
<td>varchar(11) (primary key, not null)</td>
</tr>
<tr>
<td>Passwd</td>
<td>varchar(16)</td>
</tr>
<tr>
<td>Stu_First_Name</td>
<td>varchar(20), not null</td>
</tr>
<tr>
<td>Stu_Last_Name</td>
<td>varchar(25), not null</td>
</tr>
<tr>
<td>Stu_Address</td>
<td>text</td>
</tr>
<tr>
<td>Stu_Phone</td>
<td>varchar(15)</td>
</tr>
<tr>
<td>Stu_Grad_Major</td>
<td>varchar(20)</td>
</tr>
<tr>
<td>Stu_Under_Major</td>
<td>varchar(20)</td>
</tr>
<tr>
<td>Stu_Email</td>
<td>varchar(25)</td>
</tr>
<tr>
<td>Stu_International</td>
<td>char(3)</td>
</tr>
<tr>
<td>Stu_GMAT_score</td>
<td>int(11)</td>
</tr>
<tr>
<td>Stu_TOEFL_score</td>
<td>int(11)</td>
</tr>
<tr>
<td>Stu_Adm_Status</td>
<td>varchar(15)</td>
</tr>
</tbody>
</table>
Stu_Under_GPA float
Stu_Current_GPA float
Stu_Enrollment_Semester varchar(10)

Administration Table

Administration table will contain the administrator’s information. The various attributes are:

- **Adm_ID**: varchar (primary key, not null)
- **passwd**: varchar(16)
- **Adm_First_Name**: varchar (20)
- **Adm_Last_Name**: varchar(25)
- **Adm_Room_No**: varchar(10)
- **Adm_Phone_No**: varchar(15)
- **Adm_Email**: varchar(25)

Register Table

Register table is the linking table between STUDENT and SCHEDULE tables. The creation of the linking table produces the desired 1:M relationships within the database [Rob 2000]. It contains the attributes likes:

- **Stu_ID**: varchar (primary key)
- **Class_Code**: int (primary key)
Figure 3.2 ER Diagram for GSTS

- ADMINISTRATION
  - Edits
  - 1
  - COMPLETE
    - M

- STUDENT
  - 1
  - 1
  - MAJOR
    - 1
    - has
    - 1

- CURRENT SCHEDULE
  - M
  - generates

- CLASS SCHEDULE
  - 1
  - generates

- GRADUATE CATALOGUE
  - M

- REQUIREMENT
  - 1
Current Schedule Table

Various departments offer various courses and each course can have many sections. Each section is referred to as a class and hence there is 1:M relationship between CLASS SCHEDULE and CURRENT SCHEDULE tables [Rob 2000]. Hence the schedule table contains the following attributes:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class_Code</td>
<td>int(6) (primary key, not null)</td>
</tr>
<tr>
<td>Section</td>
<td>varchar(10)</td>
</tr>
<tr>
<td>Time</td>
<td>varchar(6)</td>
</tr>
<tr>
<td>Room_No</td>
<td>varchar(10)</td>
</tr>
<tr>
<td>Adm_ID</td>
<td>varchar(10)</td>
</tr>
</tbody>
</table>

Class Schedule Table

The CLASS SCHEDULE table has 1:M relationship to the CURRENT SCHEDULE table. Each class references to only one course, but each course can have many classes. The class schedule table is mainly divided into two major fields (Accounting and Business). It consists of the information about the courses required for these two majors like, whether a course is a core course for a particular major or an elective for a major or an pre-requisite for some course. Hence, class schedule table has attributes like:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class_Code</td>
<td>int(6) (primary key)</td>
</tr>
<tr>
<td>Course_Name</td>
<td>varchar(9)</td>
</tr>
<tr>
<td>Stu_Grad_Major</td>
<td>varchar(20)</td>
</tr>
<tr>
<td>Course_Category</td>
<td>varchar(20)</td>
</tr>
<tr>
<td>Semester</td>
<td>varchar(10)</td>
</tr>
</tbody>
</table>
Graduate Catalogue Table

The GRADUATE CATALOGUE table consists of a list of courses required for Business as well as Accounting major. The graduate catalogue table is a master table and class schedule table is derived from it. It has various attributes like:

- **Course_Num** varchar(9) (primary key)
- **Credit_Hours** int(1)
- **Course_Description** text
- **Sem** varchar(10)

Complete Table

The complete table is the linking table between the STUDENT and the REQUIREMENT tables. There exists M:N relationship between the STUDENT and REQUIREMENT tables as each student can have many requirements and each requirement can be applicable to many students. So the complete table creates the desired 1:M relationship. This table also contains information about the course status, like how many courses are completed by the student and the courses that are still to be taken by the student and also the grades for each completed course. The student will not be able to register if he has not completed the pre-requisites for a course or his overall GPA is less than 3.0 or if his admission status is “Probational”. The table has the following attributes:

- **Stu_ID** varchar(11) (primary key)
- **Class_Code** int(6) (primary key)
- **Stu_Grad_Major** varchar(25)
- **Course_Status** varchar(25)
Course_Grade: char(1)
Credit_Hours: int(1)

Major Table

The students can major in one of the two fields namely Accounting and Business. Hence, the Major table will have the following attributes:

- Student_ID: varchar (primary key)
- Major_Code: varchar (primary key)

Requirement Table

The requirement table gives the pre-requisite requirements for each core and elective course. The attributes for this table are:

- Class_Code: int(6) (primary key)
- Pre_Requisites: varchar(10) (primary key)

3.3 Major Page Design

3.3.1 Login Page

This page displays a login form as shown in Fig 2.3. The system checks the information provided by the user with the account information in the database on the server side.

Input: User ID and Password

Output: If the user submits invalid information, an error message will be given. If valid information is submitted, the system will identify the user as an administrator or a student and will display the Administration Page or the Student Main Menu respectively.
3.3.2 Registration into Graduate Student Tracking System Page

This page serves to register a new user into the system.

Input: Activating the “Not a Member” link from the login page.

Output: User registration form as shown in Fig 2.4b

3.3.3 Student Main Menu

This page will display the student’s information as shown in Fig 2.5a.

Input: Correct User_ID and password.

Output: Student Main Menu.

3.3.4 Degree Plan

This page contains the student’s entire degree plan as shown in Fig 2.5b.

Input: Clicking on the “Degree plan” link in the Student Main Menu.

Output: Degree plan of the student.

3.3.5 Course Grades

This page displays the grades for all the courses registered for the semester (Fig 2.5d).

Input: Clicking on the “Grades” link in the Student Main Menu.

Output: Grades of the student for the semester.

3.3.6 Class Schedule

This page displays the entire class schedule for the semester as shown in Fig 2.5f.

Input: Selecting the “Business Class Schedule” or the “Accounting Class Schedule” from the drop box.

Output: Displays the appropriate Class Schedule for the semester.
3.3.7 Registration

Student is allowed to register online for the courses.

Input: Class codes of the courses the student wants to register for that semester (Fig 2.5g)

Output: Successful registration of the courses or an appropriate error message (Fig 2.5h).

3.3.8 Drop Course

The student is able to drop the courses online before the deadline.

Input: Class codes of the courses to be dropped.

Output: Successful in dropping the course or an error message.

3.3.9 Administration Main Menu

The Administration Main Menu displays the student’s information, where the administrator will be able to add, delete or modify the student’s information, enter the grades or edit the status of a course.

Input: Entering valid User_ID or Last Name of the student in the Find Student Page as shown in Fig 2.10a and Fig 2.10b.

Output: Administration Main Menu as shown in Fig 2.11a.

3.3.10 Edit Grades

The administrator will be able to edit grades for each student through this page.

Input: Activating the “Edit Grades” link from the Administration Main Menu.

Output: Course Grades Page as shown in Fig 2.11b.
3.3.11 Add Course

The administrator is able to enter or edit the status of the course for a student.

Input: The Course Number and its Status (“waived”, “completed”, “Permitted”) as shown in Fig 2.11c

Output: Successful entry into the database or an error message as shown in Fig 2.11d.

3.3.12 Add Student

The administrator is able to add a new student to the database.

Input: Activating the ‘Add Student” link from the Administration Page.

Output: A form is displayed to enter the information of the student as shown in Fig 2.9.

3.3.13 Enter Grades

The administrator is able to enter the grades for each course.

Input: Selecting a course from the drop box menu as shown in Fig 2.12a

Output: A screen consisting of all the names of the students currently registered for that course. The administrator can enter the grades in the textboxes provided before each student as shown in Fig 2.12b

3.3.14 Graduate Catalogue

The administrator is able to view the graduate catalogue online

Input: Selecting the ‘Graduate Catalogue’ link from the Administration page (Fig 2.8).
Output: A screen displaying a list of course numbers along with the course names, course descriptions, prerequisites and semester in which the course is offered (Fig 2.13a).

3.3.15 Print Reports

The administrator is able to print various kinds of reports to evaluate academic performance.

Input: Activating the “Reports” link from the Administration Page

Output: The corresponding page.
This chapter narrates the project implementation details. The entire project is divided into six main modules. Each module is sub-divided into several units. The following is the list of modules:

- Login Module
- Change Password Module
- Forgot Password Module
- Student Module
  - Student Degree Plan Module
  - Grades/Transcripts Module
  - Class Registration Module
- Administration Module
  - Add Student Module
  - Reports Module
  - Grades Module
  - Edit Course Status Module
  - Add Course Module
  - Graduate Catalogue Module
  - Class Schedule Module
- Logout Module

4.1 Login Module

Login Module is the first module that provides access to the Student Tracking System. *Login.php* is the file that gets the user ID and password as input from the
user. These details are then passed to the member.php file to check the authenticity of the user. The connection to the database takes place in this file. Member.php first checks if the user is a student or an administrator depending on the user ID provided by the user. On successful verification the user is redirected to the student page or the administration page. If authentication fails, a login failure message is displayed and the user is redirected to the login page. Following is the code snippet to verify the user’s authenticity:

```php
<?

// include function files for this application
require_once("bookmark_fns.php");

session_start(); //start the session

//get the user ID and password from the login page
$username = $HTTP_POST_VARS['username']; //Passing the variables from $password = $HTTP_POST_VARS['password']; // from the login input form
$username1 = $_GET['username1'];
$password1 = $_GET['password1'];

if($username1 && $password1)
{
    $username = $username1;
    $password = $password1;
}

if($username && $password)
    // the user has just tried logging in
    {
        if ($test = login($username,$password)) // call the function from output.php
            {

                // Check if the userID provided by the user is that of the administrator or student
                $check1 = mysql_query("select Adm_First_Name from administration
                    where Adm_ID = '$username'");

                $checking_array= mysql_fetch_array($check1);
```
//if the user ID is not of the administrator, then check to see if it is of the
//student
if(!$checking_array[0])
{
    $check2 = mysql_query("select Stu_First_Name from student
        where Stu_ID = '$username'".RadioButton);
    $check2_array = mysql_fetch_array($check2);
    if(!$check2_array[0])
    {
        $HTTP_SESSION_VARS['valid'] = $check2_array[0];
    }
    else
    echo "did not select any rows";
    }
elseif(!$check2_array[0])
{
    $HTTP_SESSION_VARS['valid'] = $checking_array[0];
}
$HTTP_SESSION_VARS['valid_user'] = $username;
} else
{
    // If the user tries to login in, but is unsuccessful, give an error message
    do_html_header_please("Problem:");
    echo "You could not be logged in. You must be logged in to view this
page."
    echo "<br />
    do_html_url("login.php", "Login");
    do_html_footer();
    exit;
}

//if the user enter only the userID or the password, give an error message
elseif ($username || $password)
{
    do_html_header_please("Problem:");
```php
<?php

//if the user is student then display the Student Menu
if($test == 1)
{
    do_html_header_please("Student Main Menu");
    check_valid_user();
    $new_array = checking();
    display_Stu_Menu_form($new_array);
    display_user_menu_student();
}

//else display the Administration Menu
elseif($test == 2)
{
    do_html_header_please("Administration Page");
    check_valid_user();
    display_adm_frontpage();
    display_user_menu_administration();
}

do_html_footer();
?

4.2 Change Password Module

The user is allowed to change his password by selecting the link “Change Password” from the user menu displayed at the bottom of every screen. On selecting
this link, the user is directed to the change_passwd_form.php, which displays a form to input the old password and the new password. These values are then passed to change_passwd.php where they are checked for correctness and upon successful verification the password field in the database is updated. If the verification fails, the user is given an appropriate error message and asked to reenter the values.

4.3 Forgot Password Module

The user is allowed to reset his password and set another one, if he fails to remember his old password. On selecting the “forgot password” link from the login page, the user inputs his user ID and is directed to forgot_passwd.php file, which calls the reset_password() function from the user_auth_fns.php file and the password is resetted to a random number generated by the rand() function. The notify_passwd() is then called, which connects to the database and select the email address of the user and by using the mail() function, sends an email to the user notifying him of the changed password. The user can log in using this password and then by selecting the “change password” link from the menu can set the password of his choice. For the mail() function in php to work, some changes to the php configuration file have to be made. In the config file, the localhost is assigned a domain name, which in this case is mail.sbcglobal.net. But it can be the domain name of any institution or organization like mail.tamu.edu. SMTP is used to send the email to the student.

4.4 Student Module

When the user inputs the user ID and password to log in, the member.php file verifies if the user is a student or an administrator. If identified as a student, the
The member.php file calls display_student_main_menu() from the output_fns.php file and displays the student page.

4.4.1 Student Degree plan Module

The user is allowed to view his degree plan by selecting the “Degree Plan” button from the Student page. Depending upon his graduate major, the user is directed to businessrecord.php or accountingrecord.php file which in turn makes a call to display_business_record() function or display_accounting_record() function from the output_fns.php file and displays the students record.

4.4.2 Grades/Transcripts Module

The user is allowed to view his grades and transcripts by selecting the appropriate link from the Student page. On selecting the “Grades” link, the user is asked to select a semester from the drop box and is directed to viewgrades.php file, where he is able to view his grades. On selecting the “Transcripts” link, the user is directed to transcripts.php file, where he is able to print an unofficial copy of the transcript.

4.4.3 Class Registration Module

The user is allowed to register for classes online as well as drop the classes by selecting the appropriate link from the Student page. On selecting the “Register” link from the Student page, the user is directed to a pop up window select_semester.php, where the user select a semester from the drop box. After selecting the semester, the user is directed to enterclasscode.php file, where he is allowed to enter upto five class codes for the courses he wants to register. The class codes entered by the user are checked for correctness. If any of the class codes are not entered correctly, an error
message is given and the user is prompted to reenter the class code. The user is then directed to `verifyregistration.php` file, which checks if the user has met all the requirements for registering for a course. On successful completion of all the requirements, the user is allowed to register for that course. `Verifyregistration.php` connects to the database and inputs the classcode for the course in the complete table. If the verification fails, an error message is displayed notifying the user of the reason for failed registration.

### 4.5 Administration Module

When the user inputs the user ID and password, the `member.php` file verifies if the user is a student or an administrator. On identifying the user as an administrator, the `member.php` file calls the `display_administration_main_menu()` function from the `output_fns.php` file which displays the administration page.

#### 4.5.1 Find Student Module

On selecting the “Find Student” link from the administration page will direct the user to a pop up window `find_student.php`. The user has the option of finding a student using the student’s user ID or lastname. Selecting the user ID will direct the user to `enteruserID.php` file. After the user enters the student’s user ID, `enteruserID.php` connects to the database, selects the user ID of the student from the database and calls the function `display_administration_main_menu()` from `output_fns.php` file which displays the student’s page. If the user ID is not entered correctly, an error message is displayed and the user is prompted to enter the student’s user ID again. On selecting the lastname option, the user is directed to `enterlastname.php` file. If the database contains more than one record by the same
lastname, then all the records are displayed. After the user selects a lastname from the
record he is directed to multiplelastname.php file which displays the student’s page.

4.5.2 Reports Module

The administrator is able to generate and print specific reports that need to be
sent to the registrar and other administrative offices in the university. On selecting the
“print reports” link from the administration page, the administrator is directed to
printreports.php file, which displays a list of options like students firstname,
lastname, GMAT score, TOEFL, GPA. The administrator is allowed to choose one of
the options and is directed to sortedreport.php, which displays the report that is sorted
using the option selected.

4.5.3 Add Course Module

The administrative user can add a course to the student’s record by selecting
the “Add Course” link from the Administration Main Menu page. On selecting this
link, the user is directed to addcourse.php file, which displays a form for the user to
input upto five courses and their status (“waived”, “completed”, “registered”,
“permitted”). After submitting the information, the user is directed to
addedcourse.php file, which checks if the course numbers and their status entered is
correct. If the verification fails, the user is notified by an appropriate error message
and prompted to enter the information again. On successful verification, the course
and the status are entered into the complete table in the database.

4.5.4 Grades Module

The administrator is able to enter the grades for the students online by
selecting the “enter grades” link from the Administration page. On selecting the link,
the administrator is directed to a pop up window selectsemester.php where the administrator has to select the semester for which he wants to input the grades for the students. The administrator is then directed to entercourse.php file, which displays all the courses offered in that semester and the administrator is allowed to select one course from the drop box. After selecting a course, entergrades.php file is called which displays all the names of the students that are currently registered for that course. When the administrator enters the grades for each student, the student’s record is automatically deleted from the “register” table and inserted into the “complete” table. The entergrades.php file then calls the calculateGPA() function which calculates the G.P.A for each student and stores it in the student table.

4.5.5 Class Schedule Module

The administrators as well as the students are able to view the Class Schedule online by selecting the “Class Schedule” link from their respective pages. Two class schedules are available online – business class schedule and accounting class schedule. On selecting the semester and one of the options from the drop box, the user is directed to either stuclassschedule.php or admclassschedule.php, which displays all the courses that are offered in that semester along with their timings, room number, pre-requisite requirements and instructor’s name. If logged in as a student, the user is just able to view the schedule, but if logged in as an administrator, the user is also allowed to add, modify or delete a course from the class schedule. When the administrator selects the “Insert a Course” button, he is directed to insertrecord.php, which displays a form to fill out the information of the course. When the administrator fills out the form and selects the “Save” button, the
insertclassschedule.php file is called which checks if all the information filled out in the form is in the correct format. The administrator is only allowed to insert a course, which is already present in the graduate catalogue. If not, an error message is displayed, else the course is inserted into class schedule table. If the administrator chooses to delete a course from the class schedule, he can do so by selecting the “Delete” button. Upon the selection, he is directed to actiononschedule.php file. The administrator is asked if he wants to delete the selected course. If he clicks the “No’ button, he is directed back to admclassschedule.php. If he decides to delete a course and chooses “Yes”, he is directed to saveclassschedule.php file which will delete a course from the class schedule table, but not from the graduate catalogue. The administrator also has an option of editing a course. When he selects the course to be edited and clicks on the “Edit” button, he is directed to actiononschedule.php file which displays a form containing the current information of the course. If the administrator wishes to change the information, like the timings of the course or the days on which the course is offered or the name of the instructor who teaches that course, he is able to do it, and by clicking on the “Save” button, he is directed to saveclassschedule.php file which updates the database with the new information.

4.5.6 Graduate Catalogue Module

The administrator is able to view the graduate catalogue online by selecting the “Graduate Catalogue” link from the Administration Page. On selecting this links, the maincataloguesem.php page is loaded where the administrator is allowed to choose a semester. When the administrator selects a semester, he is directed to graduatecatalogue.php file, which displays the graduate catalogue. If the
administrator wants to add a new course to the catalogue, he can do it by selecting the “Insert Course in Catalogue” button, which loaded the insertmaincatalogue.php file, which in turn calls a insert_maincatalogue() function. This displays a form where the administrator is required to fill out the information of the new course and on selecting the “Save” button, he will be directed to insertmaincatalogue.php where the information is stored in the graduate catalogue table. When the administrator selects the “Edit” button, he is directed to actiononmain.php where he can change the information regarding the course, like name of the course or description of the course or semester the course is offered or the prerequisites required for that course etc. On clicking the “Save” button, the editmaincatalogue.php file is loaded which updates the graduate catalogue table as well as class schedule table. Similarly, the administrator is able to delete a course from the catalogue by selecting the “Delete” button. This takes the administrator to deletemaincataloguewindow.php, where the administrator is asked whether he wants to delete the course. If he chooses the “No” button, he is taken back to graduatecatalogue.php and if he selects the “Yes” button, he is directed to deletemaincatalogue.php, where the course is removed from the graduate catalogue as well as from class schedule.

4.6 Logout Module

This module logs out the user from the system. The logout.php file has the PHP script to logout the user from the system. Once the user selects exit option, all the cookies are unset and the user is forbidden to access the system. The user is permitted to use the system with a login again. This module redirects the user to the login page.
5. TESTING AND EVALUATION

5.1 System Performance Metrics

The Graduate Student Tracking System is a successful Web application. The system particularly helps the graduate coordinators in tracking and maintaining the student’s academic information. This system helps users achieve their goals quickly and efficiently. It is designed and implemented to achieve the following objectives:

- It addresses the needs of the user
- It is easy to use
- It delivers content that is correct, complete and current
- It is easy to maintain

5.2 Software Testing

Software testing involved the functional performance check. It has been tested to check if the tool performs all the functions. The system is tested while it is being developed. The project is divided into modules and each module, in turn, is divided into many units. Several distinct test cases are evolved for each unit as well as for each module and finally for the entire tool. Both black box and white box testing are adopted. A coding standard is adopted and proper documentation of the code is followed for ease of inspection and maintenance.

Entire system is divided into modules. Independent and integrated testing are performed at every stage of development. Each unit is developed and tested simultaneously. Once the units of a module are tested independently, the units are combined to form the module. Each module is tested independently for data integrity,
data consistency, consistency of data input and output formats and for other features. Each module is tested individually to check if it has achieved the objectives of the set of requirements. Once all the modules are developed, the modules are combined and tested.

5.3 Usability Testing

There are various methods of testing and evaluating the Web site. During the initial stages, the developer tested the Web pages. The developer tested the pages while adjusting the browser setting as follows:

- Turning off images, and making sure that the information is presented in an appropriate sequence relative to the visual presentation on the GUI site [Nielson 2000].
- Changing the font size (larger and smaller) in the browser, and observing whether the page is still readable.
- Setting the screen resolution to 640 X 480 and observing whether or not this forces the page into horizontal scrolling.
- Changing the display color to black and white and observe whether color contrast is adequate.
- Putting away the mouse and accessing the links and form controls with the keyboard.

The download times for the Web site using different Internet connections were also taken into consideration while designing the Web pages. Emphasis was given on fast download time. The Web pages were also tested using different kinds of browsers such as Netscape Navigator, Internet Explorer.
Later after the tool is completely developed, Dr. Sameer Vaidya, an Associate Professor and a graduate coordinator in the College of Business at Texas A & M University – Kingsville, was asked to evaluate the usability. Suggestions from the graduate coordinator have been incorporated for better usability of the system. The Web pages are tested on the developer’s laptop computer and the Web site usability is evaluated for the following:

- Ease of finding the specific information
- Ease of reading the data
- Logic of navigation
- Timeliness of data (is it current?)
- Completeness with which the site’s subject is treated
- Appearance of the site
- Web site layout consistent
- Ease of search
- Explanations of how to use the site
6. RESULTS AND CONCLUSION

The result of this project is a Web-based Graduate Student Tracking System with security features. The system is greatly helpful to the graduate coordinators in tracking each student’s information right from the time of their application to the program to the completion of their degree. The graduate coordinator will no longer have to depend on the University’s Office of Admissions to get the information relating to his graduate students. The coordinators are also able to enter the grades of the students online and pull out various kinds of reports for analysis. As it is an online system, it is also useful to students. They are able to change their personal information, check timetables and grades and view their entire academic record and get an unofficial copy of the transcript. The system is supported by a relational database and provides a consistent and friendly Web interface to allow the users to view desired information, perform exercises, query the database and generate reports.
7. FUTURE WORK

There is a scope for future work with this project. This tool can be upgraded with additional features. Some of them are listed here:

- The Graduate Student Tracking System is a standalone system. But it can be connected to the university’s main registration system, so that the graduate coordinators can stay current with the student’s progress with regard to their degree requirements, print various kinds of reports for departmental level decisions and can change the information of the student, if required. The students will also be able to benefit from this system as with any other online registration system.

- Functions can be written to produce visual graphs after the coordinator obtains a sorted report. For example, the student table already has a field ‘Semester_of_Enrollment’ for all students. If the coordinator wants to sort the report with the GPA’s of all the students with semester of enrollment from, say Fall 1998 to Fall 2004, he should also be able to get a graphical representation of the report. Graphs provide better visual evidence of the trends and are easier to interpret.

- In future, if the university decides to add a PhD program to the College of Business, the system should be updated to track the PhD students as well.
ACKNOWLEDGEMENTS

I would like to render my sincere thanks to Dr. John Fernandez, my Project Committee Chair for his constant support, valuable insights and guidance throughout the project. I sincerely thank Dr. David Thomas and Dr. Chandrika Rao for being on my project committee.

I am thankful to my husband, Dr. Sameer Vaidya for sparing time to evaluate this project and for his feedback about the usability of the tool. I would also like to thank Mr. Marcial Reza, Library Assistant II, at Bell Library for his input and suggestions.

Above all, I thank the Almighty for giving me the strength and will to finish the undertaken task, successfully.
BIBLIOGRAPHY AND REFERENCES


