Prototype Web-based Database for Student Registration System

Siva Kumar Kuruganti[1]
CACI International Inc.
639 Loyola Ave, Ste.2550, New Orleans, LA 70113

and

Ghasem S. Alijani[2]
Graduate Studies Program in CIS
Southern University at New Orleans
6400 Press Drive, New Orleans, LA 70126

Abstract

Student registration systems that are in place at many universities require students to come to the campus and register for respective courses. Such systems put strain on the computing and human resources available at educational institutions. Some of the problems that are inherent to these systems include time consumption in the registration process, pooling of computing and administrative departments, and an acute increase in the staff requirements to complete the registration process in a short span of time for all students. This project concentrated on the development of a Web-based student registration system that would alleviate the inherent problems by providing an efficient, fast, and effective registration process that is accessible through the Internet. Several advantages can be realized from this prototype. All the administrative departments involved in the registration process can integrate their functions to provide a comprehensive process that would enable the student to register for courses, pay the fees, and get the financial aid based on the eligibility. The prototype developed in this project provides an additional option of posting grades of the students on the University's Web site, thus enabling the students to view their grades without having to come to the campus or access any cumbersome telephone-based processing systems. The full-scale model of this prototype will support students, administrative staff, and faculty of any educational institution in managing their time and resources effectively through on-line communication.

Keywords: Student registration, prototype, web-based system, open database connectivity, active server pages

1. INTRODUCTION

The World Wide Web is fast becoming the most popular way of accessing the Internet, as it is easy to use graphical interface and the ubiquitous Hypertext Transfer Protocol (HTTP) for communication. An organization makes itself accessible to Web users by maintaining a home page on a Web server that can be accessed from any other location worldwide using a Uniform Resource Locator (URL).

Business applications require a Database Management System (DBMS) for storage and retrieval of the organization's valuable data. There are various business applications in a university for which data is stored in a database. Some of the applications are student data, registration information, grades, fee payments, faculty data, course information, etc. Most of the information related to a university is accessible only from the documents or catalogues available in the school.

Web-to-database interfacing provides a Web page with interaction: the ability for the Web to go beyond a computer-based publication medium into the domain of feedback, response, assessment, and choice. There are two main approaches that allow Web-based input forms written in HyperText Markup Language (HTML) to communicate with a database. They are Common Gateway Interface (CGI) and Server Software.

[1] skuruganti@hotmail.com
[2] dalijani@ix.netcom.com

The CGI is the most proven method, but requires complex coding to implement. It is a standardized exchange of information between four components: (1) the user's Web browser (client), (2) the server, (3) a CGI program (script) and (4) the database. Most Web server software is compatible with the CGI standards. Unlike CGI programs, server software is an extension of the Web server software, adding functionality and ability to interface with databases. Personal Web Server (PWS) and Internet Information Server (IIS), two server software products of Microsoft Corporation, are very popular with the Windows operating system.

The registration process at the university is mapped and is shown in the form of a node tree in figure 1. Delays occur at the following points during the registration process.

- At the advisor's office for getting advised as to which courses the student has to register.
- At the registration terminal for entering into the system.
- For taking the printout of the registered courses to know what fees have to be paid.
- At the financial aid office.
- At the fee deferment office.
- And finally at the cash counter.

These delays make the registration process time-consuming.

Many students register each semester for various courses during a short period of time that may result in operational and logistical difficulties. Some of the operational problems may include pooling of computers, moving people from various departments, ensuring connectivity of all the computers to the mainframe, and training some of the temporary staff. Such adjustments at the time of registration put strain on all types of personnel and financial resources.

The objective of this research project was to design and develop a prototype Web-based registration system for a University. Prototyping reduces the number of problems with requirement specifications and the overall development costs. Effective prototyping increases the software quality. There is a reduction in the development time by about 40 percent and an increase in user satisfaction by 65 percent when the production-quality system is built from the prototype (Bernstein 1993). With a Web-based registration system students can register from virtually anywhere and need not come to school for registration. This will greatly reduce the amount of time the student spends to complete his or her registration, and from the school's point of view, it will reduce the strain on already limited resources. The advantages of prototype are detailed below.

1) A prototype is close to the actual system to be built because the prototype is developed through an iteration process, with the user involved for validating the functionality of a system.
2) Prototyping accelerates several phases of the development life cycle.
3) An approved prototype is equivalent to a paper design specification.
4) Errors can be detected much earlier in a prototype than when the system is actually built.

2. LITERATURE REVIEW

Storey et.al (1997) state that even when working in a relatively unfamiliar application area, the database designer is usually able to make helpful suggestions and inferences while recognizing possible errors in the user's input. The designer applies common sense reasoning and general knowledge about the world to the user's problem. For example, if the user mentions a university, a human database designer immediately suggests the need for entities such as Student, Course, and Department. This is translated into an automated database design tool to incorporate knowledge and reasoning capabilities to support a higher level of participation. Common Sense Business Reasoner is a prototype that was implemented by using general world knowledge about business for database design. The system interacts with a user using a set of menus to classify the user's elements and computes the distance function values. The authors described a database for a student advisory function.

Nguyen and Srinivasan (1996) gave an overview of the 'DB2WWW Connection' and how it runs. An interface is needed between HTML and the Structured Query Language (SQL) used to access the relational DBMS. The authors gave a mechanism called DB2WWW Connection that provides cross-language variable substitution between HTML and SQL and helps in quick and easy construction of applications that access relational DBMS from the Web. The authors developed their own syntax for cross-language variable substitution between HTML and SQL. They discussed variable-assignment, declarations, executable variable declarations, SQL report block, HTML input section, and HTML reports section.

White (1997) reported on the problem of storing large amounts of information created, received, and processed every week by different departments in a university. The author suggests that Web-to-database interfaces fit in a campus service where large numbers of users require answers to the same questions. In this case a knowledge database that users can query can be set up on the Web. White further explains about CGI and server software that help the Web site to communicate with the database.

Chen and Rishe (1998) reported the design principles of the Web-based information publishing system that was developed at Florida International University. The authors presented information about facilitating efficient querying of databases through the Web and rapid application development. They tested several design issues related to performance and functionality.

Claverie (1998) discussed the use of IIS and Visual Basic (VB) ActiveX Controls. He described the relation between the client/server model and the Internet and how an IIS allows users of the Internet to query remote databases. An advantage of IIS is that the database is stored on the server and only authorized users can have access to the database. The advantage of using VB is the short span of time in which ActiveX applications can be generated and executed.

3. METHODOLOGY

The following steps were followed in the development of the prototype for the Web-based registration system:

1) Identification of objects that need to be in the registration system from various sources such as course bulletins, catalogue, and journals.
2) Capture of various business rules.
3) Development of Entity-Relationship models (logical and physical) that show data structures, associative tables, and views.
4) Creation of a database for the registration system using Microsoft SQL Server 7.0.
5) Installation of a Web server and Microsoft FrontPage 2000 for the development of Web pages.
6) Establishment of connectivity between FrontPage, the Web server, and the database.
7) Development of Web pages for the registration system.

The first step in the development of the Web-based prototype was to identify the objects that are involved in the registration process. The objects are identified from application for admission, course registration form, grade report, and course bulletin. The objects involved in the student registration process are advisor, student, courses listed in the courses bulletin, instructor, and the rooms in which courses are taught. At the same time, the attributes of each of these objects have been identified. Table 1 shows the identified attributes for some of the entities, and a comprehensive entity-information can be obtained from figure 2.

Table 1. Entities and the related attributes

Entity	Attributes
Student	student Id, student name, sex, date of birth, mailing address, telephone number, email address, student classification, major, student enrollment status, courses registered, grades in each course, advisor's name
Instructor	instructor Id, instructor name, address, telephone number, email address, courses being taught, students to whom the instructor is advising
Courses	call number, course Id, course title, credit hours, activity type, prerequisites, where the course is taught, which instructor is teaching it, section number, timings, days on which the course is taught, semester in which course is offered
Rooms	room number, room description, number of seats available, and which building

The second step involved normalization of the attributes to remove redundancy and the entities were normalized to the third normal form. From the normalized entities and their associated attributes, entity-relationship diagrams were developed. An entity-relationship diagram shows the entities, their attributes, and the relationships between them. Some of the integrity rules are identified through relationships between different entities. The entities after normalization are STUDENTS, FACULTY, COURSES, ROOMS, COURSES_OFFERED, and STUDENT_COURSES. The relationships in the registration system are:

1) Many students can register in many courses.
2) A faculty member can teach 0, 1, or several courses and a course must be taught by one faculty.
3) From the total courses offered by the university, 0, 1, or several courses are offered in each semester, and a course offered in the semester should be in the list of courses.
4) A room is allocated to 0, 1, or many courses and a course can be taught only in one room.
5) A faculty advises 0, 1, or many students, and each student is associated only with one advisor.

There is a many-to-many relationship in the registration system between STUDENTS and COURSES. This relationship is broken into many-to-one and one-to-many with the help of an associative entity, namely STUDENT_COURSES.

The physical entity-relationship diagram, which is developed from the logical E-R diagram, shows the actual implementation of the E-R diagram into a database. It gives the structure of the tables, the size of each attribute and its data structure, views that are

generated in the database, and their relationships. The physical E-R diagram that is developed from the logical E-R diagram and used for development of the registration system is displayed in figure 2.

The next step in development of the prototype was the creation of the actual database in MS-SQL Server 7.0. Using the structured query language known as Query Analyzer, tables, views, procedures, and triggers were created. Some of the business rules were captured while creating the table. For example, while creating the COURSES table, ACTIVITY TYPE can only be one of the following: 'LEC', 'LAB', 'IND', 'DIS', 'FLD', 'INT', 'RSH', 'QUZ', 'STU', 'REC', 'SEM', or 'ACT'.

Triggers and procedures also were used, which help in capturing the business rules. Triggers are executed when conditional events associated with them occur. For example, student classification should change depending upon the total credit hours completed by a student.

4. SYSTEM ARCHITECTURE

Client/server systems are constructed so that the database can reside on a central computer known as a server and can be shared among several users. Users access the server through a client or server application.

- In a two-tier client/server system, users can run an application on their local computer, which is known as a client, and this connects over a network to a server running SQL Server software. The client application runs both the business logic and the code needed to display output to the user.
- In a multi-tier client/server system, the client application logic is run in two locations: (1) The client application is run on the user's local computer and is focused on displaying results to the user and (2) The business logic is located in the server applications. Clients request functions from the server application, which in itself is a multithreaded application capable of working with many concurrent users. The server application is the one that opens connections to the database server and can be running on the same server as the database, or it can connect across the network to a separate server operating as a database server.

Since this project was developed on a desktop, both the client application and SQL server were installed on the same desktop. With the help of a client application the user requests services from the SQL server, which also is on the same machine. The server with the business logic retrieves data from the database and sends it to the client.

To develop a Web-based database on a desktop using the SQL Server, the PWS has to be installed on the desktop. PWS is a desktop Web server that helps in establishing a Web site setup. Personal Web Server supports Active Server Pages and can be used as a development and testing platform for Web sites. PWS is suitable for personal publishing on corporate Intranets. However, PWS has a 10-connection limit, which means that it cannot support more than two or three visitors at a time, so it is not suitable for direct Internet publishing. The system architecture with a PWS on a desktop computer, which has the SQL Server, is shown in figure 3. This is the system architecture followed for the prototype Web-based registration system.

A Web site is a location on a computer network that provides information in the form of pages or documents that are available to visitors when they reach the site with the browser software. The computer network can include the Internet or a company's Intranet. The information can be published in the form of HTML pages or in other document formats. To view the information available on the site, visitors use browser software programs, such as Internet Explorer or Netscape Communicator that translate HTML pages on Web sites to text and graphics on their monitors. Several HTML pages are bundled together to form a Web site.

The database integration is based on Active Server Pages (ASP). ASP is an HTML document that contains embedded server-side scripting. Web servers that are compatible with ASP can execute the server-side scripts. On the client side, an ASP is a standard HTML document that can be viewed on any platform using any Web browser. Therefore, all the pages that interact with the database were created on the server as Active Server Pages.

Before the information in the registration system is used in the Web pages, a database connection has to be established. The connection used for the prototype registration system is based on System Data Source Name, which uses ODBC. With the help of ODBC drivers for MS-SQL Server, a connection is established between FrontPage 2000 and the registration system database by giving the name of the database, the user's name, and a password. A connection once established will always be there unless deleted. A new connection to an existing or a new database can be established without deleting the existing connection.

From the Web page, a student can query the database to view all the courses that are offered by the university or can view the course bulletin. Faculty can query the database to get a list of students from a particular class or a report on grades given to students in a particular class. A faculty member can insert his/her details from the Web pages to the FACULTY table. The details about a faculty member also can be known with a search option, which searches the database for the name of the faculty member entered by the user. Students can insert their details into the STUDENTS table. They also can register for the courses. If the call number of a course is

not known, a search for the call number can be made in the same registration form with the help of a course Id. The pages also have capability to search a student by Id or name. Students can view or print the courses taken by them after registration and can also view or print their grades by choosing the appropriate option on the Web page.

5.   CONCLUSIONS

In the traditional registration process, students and staff have to spend much time in the registration process. Also, there is a great strain on resources for the school. Software prototyping will help in developing and checking the requirements of the final system.

The results of the current project are listed below.

1) A full-fledged Web-based system will save time for students because they can register from virtually any place if they have Internet access. After the development of a full-scale registration system, there is no need for the student to come to the school for registration.
2) Students can view courses offered during the semester and they also can receive their grades over the Internet through a Web-based database.
3) The faculty can save time in advising students because they can acquire a report of the students who are registered in their respective departments, and check to determine if the courses taken by the student are appropriate.
4) The faculty can acquire a report on the size and composition of a particular class.
5) This prototype has the capability to allow a faculty member to post grades for students through a Web page provided for that purpose.
6) A Web-based system can reduce the strain on the staff and other resources involved in the registration process, and the resources can be diverted for efficient utilization in other units.

A prototype is a functional representation of a design, in this case, a Web-based database for a registration system. The development of a prototype for a Web-based database for a student registration system will serve as a stepping-stone for development of a full-fledged Web-based registration system.

The prototype has to be validated by various users and maintainers of the database before it can be implemented as a full-fledged Web-based registration system. Extending the prototype to other departments and adding other processes to the registration process is recommended before developing a full-scale registration system. Also, a provision for external connectivity/interface with credit card companies and banks for authorization/approval of on-line credit card transactions can be implemented.

6.   REFERENCES

Bernstein, L., 1993, "Get the Design Right!" IEEE Software, 10 (5), pp. 61-63.

Chen and Rishe, 1998, "Development of Open and Scalable Web-based Information Publishing System." Proceedings of the 36th annual conference on Southeast regional conference, pp. 163-165.

Claverie, 1998, "Working the Net: Developing Applications with the Internet Information Server and Visual Basic ActiveX Controls." Proceedings of the 36th annual conference on Southeast regional conference, pp. 158-162.

Nguyen and Srinivasan, 1996, "Accessing Relational Databases from the World Wide Web." SIGMOD '96, Proceedings of the Conference on Management of Data, Montreal, Canada.

Storey, C., G. Dey, and Sundaresan, 1997, "Database Design with Common Sense Business Reasoning and Learning." ACM Transactions on Database Systems, 22 (4), pp. 471-512.

White and Davis, 1997, "Automating Information Overload: Linking Databases to the Web." Proceedings of the 25th SIGUCCS Conference on User Services: Are you ready?
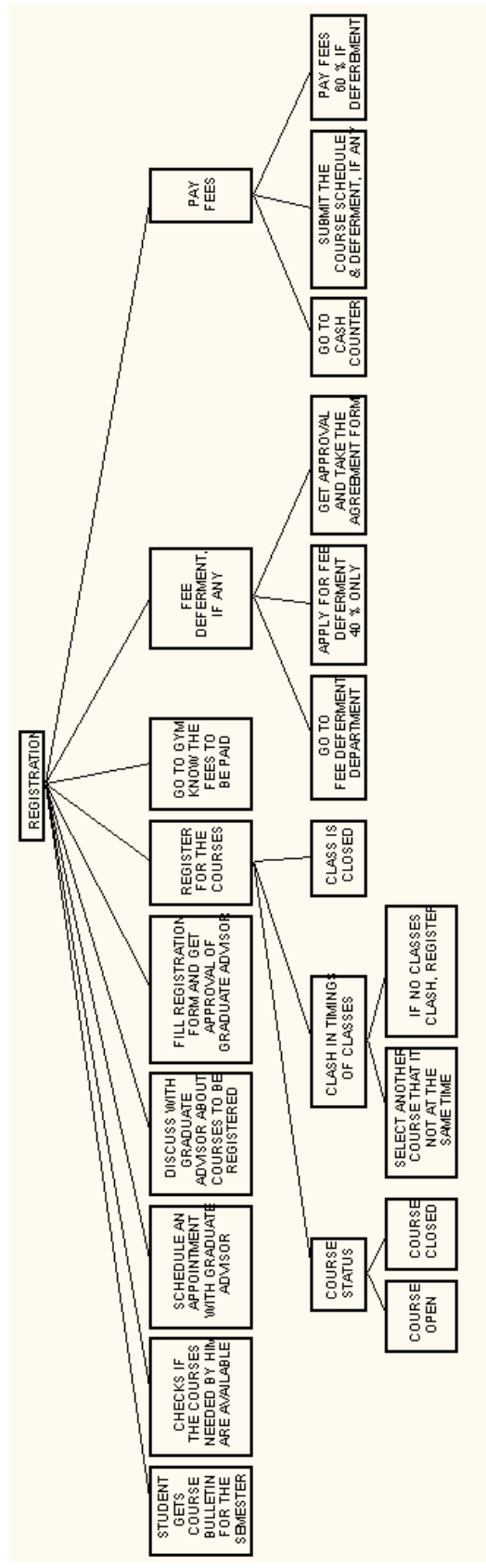
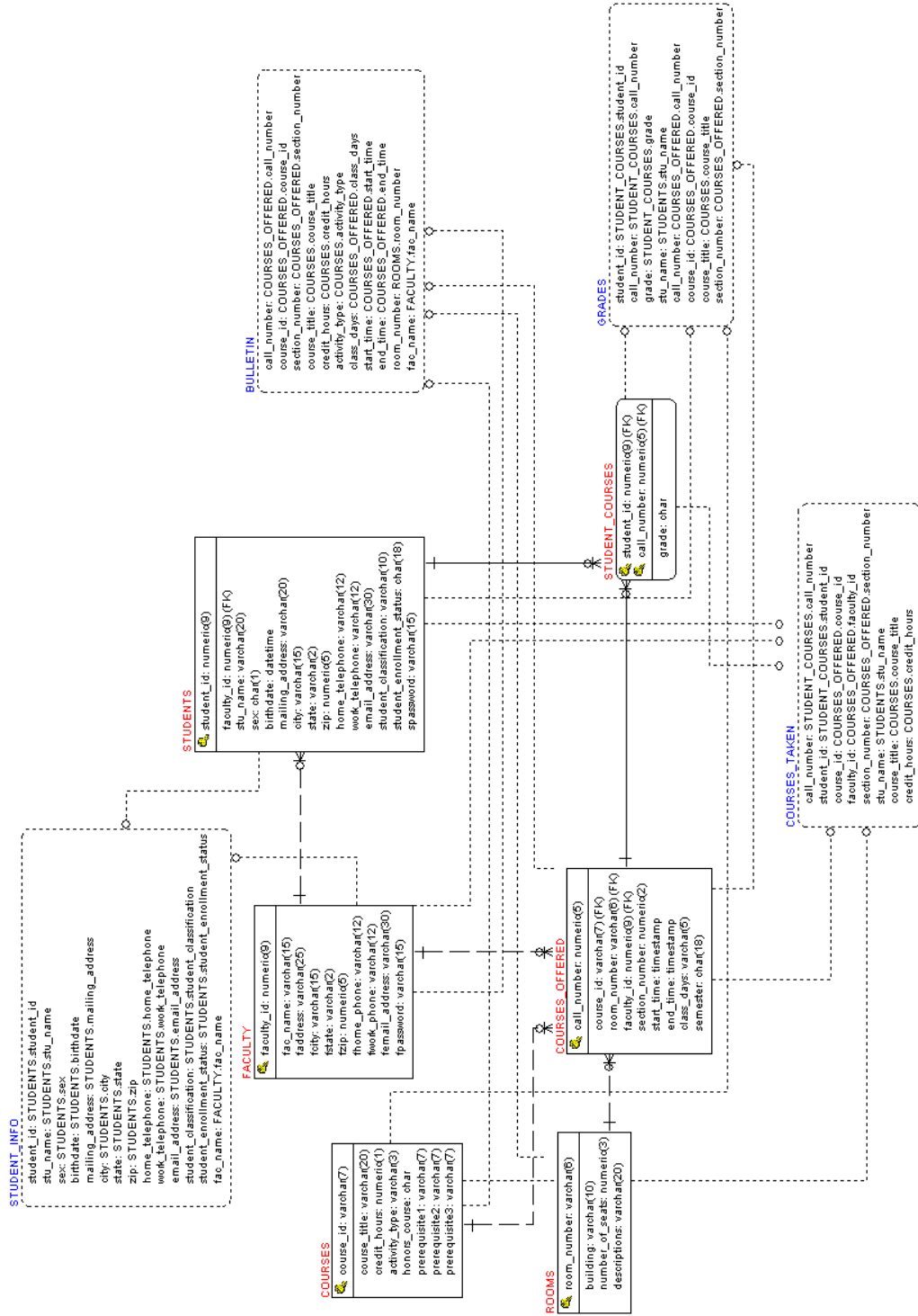Figure 1. Schematic of the Traditional Registration Process

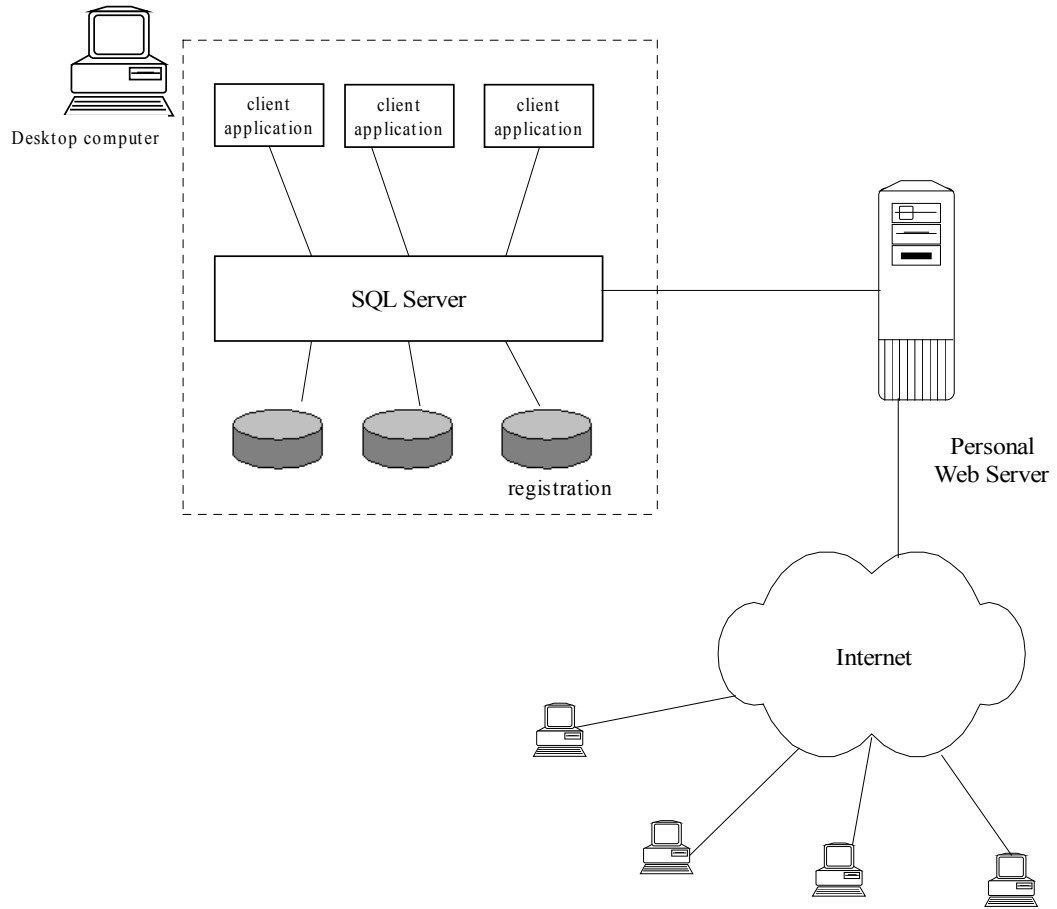Figure 2. Entity-Relationship Diagram for Student Registration System

Figure 3. System Architecture for the Web-based Registration System