Volume 13, No. 3
May 2015
ISSN: 1545-679X

Information Systems Education Journal

In this issue:

The **Information Systems Education Journal** (ISEDJ) is a double-blind peer-reviewed academic journal published by **EDSIG**, the Education Special Interest Group of AITP, the Association of Information Technology Professionals (Chicago, Illinois). Publishing frequency is six times per year. The first year of publication is 2003.

ISEDJ is published online (http://isedjorg). Our sister publication, the Proceedings of EDSIG (http://www.edsigcon.org) features all papers, panels, workshops, and presentations from the conference.

The journal acceptance review process involves a minimum of three double-blind peer reviews, where both the reviewer is not aware of the identities of the authors and the authors are not aware of the identities of the reviewers. The initial reviews happen before the conference. At that point papers are divided into award papers (top 15%), other journal papers (top 30%), unsettled papers, and non-journal papers. The unsettled papers are subjected to a second round of blind peer review to establish whether they will be accepted to the journal or not. Those papers that are deemed of sufficient quality are accepted for publication in the ISEDJ journal. Currently the target acceptance rate for the journal is under 40%.

Information Systems Education Journal is pleased to be listed in the 1st Edition of Cabell's Directory of Publishing Opportunities in Educational Technology and Library Science, in both the electronic and printed editions. Questions should be addressed to the editor at editor@isedj.org or the publisher at publisher@isedj.org.

First Database Course – Keeping it all Organized

Jeanne M Baugh
baugh@rmu.edu
Robert Morris University
Moon Township, PA 15108

Abstract

All Computer Information Systems programs require a database course for their majors. This paper describes an approach to such a course in which real world examples, both design projects and actual database application projects are incorporated throughout the semester. Students are expected to apply the traditional database concepts to actual database storage problems. The design of the database is emphasized and students evaluate each other's design as well as their final database project. The details of a database written by the Author to organize all information surrounding the student's project are also presented. In this database course, the students are able to follow the progress of their fellow students during the course. Student's evaluations of each other's work during the semester is shown to be beneficial in the learning process.

Keywords: Database first course, Real world projects, Database project, IS education

1. INTRODUCTION

All students in an Information Systems degree are usually required to take one database course as part of the core for the major (IS2010, Model Curriculum and Guidelines-http--www.acm.org-education-curricula). Such a course would emphasize database design and implementation based on a thorough analysis of requirements and information modeling. An introduction to relational database technology would be provided, highlighting the use of Structured Query Language (SQL) and report generation. An advanced course could go into more detail in the areas of advanced SQL considerations, PL/SQL, database performance and security issues, multimedia, parallel, and distributed database management systems, data warehousing, and object-oriented databases. At the Author's institution, the undergraduate student takes only the first, beginning course.

In the database course described in this paper, students are given as much real world experience as possible. Therefore, a great deal of time is spent in design work of simulated real world problems as well as a detailed design of the student's individual semester project. If the course material can be made more interesting to the student, then he will be more inclined to want to learn it. Real world projects allow the students to "learn better through a particular domain of their interest" and "see the practical value of what they learned" (Robbert, Wang, Guimaraes & Myers, 2000). Others have had success with a "real world" approach, such as the partnership of a university with a large insurance company to implement database assignments for introductory and advanced database courses (Seyed-Abbassi, King & Wiseman, 2007). To give students such experiences in the "real world", a bridge must be established between business and academia (Courte & Bishop-Clark, 2005). In teaching beginning database for many years, it has been the Author's experience that students have a relatively naive view of the complexity of real-world business.

Additionally, students are expected to also communicate with actual users. (Baugh, Davis, Kovacs, Scarpino & Wood, 2009). Database programming skills along with communication skills go hand in hand for successful employment (Seyed-Abbassi, King & Wiseman, 2007). In fact, employers do demand this of their entry level employees (Gruba & Al-Mahmood, 2004). Thus, large semester projects along with documentation should be key elements of any database course (Ehie, 2002). And, gaining practical experience is invaluable when the student is on that first job search.

Database Course Emphasizing Design Work

Two different approaches to teaching a database course could be used. One focuses on the actual application software, Oracle, SQL Server etc… The other focuses on the theory of database design and implementation, with the vehicle (the database package) being secondary. This paper describes the later approach. A database can not be implemented until the design is clean (Deperlioglu, Sarpkaya & Ergun, 2011). Or perhaps better stated, it can not be implemented successfully if there are normalization problems.

This Author has worked as a database consultant for many years, with the most recent projects being complicated databases in the orthopedic department of a large city hospital. Many of the Author's experiences are brought into the classroom. This allows the students to see what a database designer actually deals with in designing and implementing database projects for real users. Students are also given actually database problems from live medical databases (with data scrubbed out, of course).

Database design is a major element of the course work described in this paper. The students were told on the first day of the course that design would be the hardest thing they would do during the course (Connolly, & Begg, 2005). After a few weeks, the students did believe this! It is the Author's opinion that many of the database text books do not spend enough time teaching the actual design of a database (Philip, 2007). In this course, the students spend at least a month on database design only, before they actually work with any physical database in the computer.

The students are provided with many examples of interactions with users (Hansen, 2012). Communication skills of the database designer are extremely important (Codd, 1971). Students are taught to "listen" to the user. Students not only need to hone their technical skills, but they need to develop a communication expertise that will allow them the best possible chance at drawing out of the user what their data needs may be. Often the users will not know what to tell the database designer about their data needs. Sometimes they will even try to tell the designer what actual fields should be in a table. They will speak in terms of the outcomes of the database, for example the desired reports. Students are taught to take those required reports and "back design" the database from them. They are also taught to ask questions (as many as possible) to help understand how all of the data fits together. For example, if they are designing a surgical database, and they are told that the Doctor wants to store the type of anesthesia used in the surgery, the student should then ask if there can be more than one anesthesia administered during the surgery. The students are provided with many examples of user issues and some "mistakes" the Author has had to deal with in designing and implementing databases at the hospital. The "anesthesia" is one mistake this Author made on an actual database. By not accounting for multiple anesthesia, the students saw that a mistake can cause a major database table change. They were able to appreciate how difficult it was to deal with this mistake after 6 months of data had been entered. Seeing how actual design errors cause very big problems is a valuable tool. By knowing what the user expects from the database, the design can be adjusted accordingly before data has ever been entered.

Although database text books contain design assignments and projects, this Author has written many the students work through. The kind of text that is written in these design assignments has come from this Author's many years of dealing with users and helping them define what their needs really are. The Author acts as the user in these scenarios and students must ask as many questions as possible of the user to understand what the design should be. A few examples of these design problems are listed below. (The students were to create the Entity Relationship diagram for each problem). An entity is a distinguishable item and a relationship is the connection between the sets of entities.

a. You and your family have an internet business selling items on EBay. You are having some problems keeping track of

which items belong to which individual. You have been asked by your family to organize the EBay records. (some of your items are also consignment items from friends and others) Associated with the sale of the item you will have shipping information, including the customer who purchased the item. The reports you will need include:
- All information on items sold within a specific time frame
- Which family member (or friend) is selling the most items
- All open auctions
- List of all shipping costs
- Newsletter to all customers
- History of the bids on each item

b. You are the coach for a youth soccer team. Your players are children between the ages of 6 and 8. You have both male and female members on your team. You not only want to keep track of goals scored and who the opponents were, you must also know who played in the games, because the players will get a trophy if they attend all 10 games. You must also keep track of who is allowed to pick up each child after practice (the guardian). (Every individual guardian must be listed for each player) The reports you will need include:
- List of all players and their guardians
- Record of the games won and lost along with the opponent
- List of all goals each player scored and against which opponent
- List of all players attending each game
- Mailing list for the end of year banquet

The students used these and many other simulated problem statements to work through Entity-Relationship (E-R) diagrams. Often the students worked on these problems in groups. The solution E-R diagram for each was then provided to the students. The youth soccer solution is provided in Figure 1. in the appendix.

A number of assists/or shortcuts have been developed by the Author to help the student to create the E-R diagrams. For example, the students are taught to listen for any plural words in the problem statements. (a patient has **surgeries**, a student is enrolled in **courses**, an insurance agent has **clients**) When a plural exists, that is an indication that the entities should not be stored together. The E-R diagram for the patient and surgeries is in Figure 2 in the appendix.

Another assist the Author has presented concerning the E-R design is a somewhat easy way to define the cardinality. In the example in Figure 2 in the appendix, cardinality is the relationship between the patient entity set and the surgery entity set. The relationship is a diamond in the diagram and in this diagram it is called "Has". Is the relationship one-to-one, one-to-many, or many-to-many? Defining the type of relationship is often very difficult for the first time database student (Seyed-Abbassi et al., 2007). The students are taught to start with one of the entities in the relationship and ask if each of those entities can have multiple connections on the other side. If they say Yes, then an "M" is placed on the other side. Then they move to the other side of the relationship and ask the same question in the opposite direction. Using the E-R diagram in Figure 2, the method is applied as such:

Start with the Patient entity set and ask "Can each patient have multiple surgeries"? If the answer is YES, then an "M" is placed next to the surgery entity set. Then go to the surgery set side and ask, "Can a surgery be related to more than one patient "? In this case the answer is NO, so a "1" is placed next to the Patient entity set. This method will work, no matter which entity set you may start with. This is not the notation database books use for cardinalities, but in learning to create the designs, students do well with this method. Later in the semester the standard notations are discussed (Foltz , O'Hara & Wise, 2004) (for example, (1:N) or (M:N) or (1:1)). Students are also given examples of multiple relationships. One example of this is the special case when one of the related entities participates as 1 and the others as N.

After the students have had about a month of design work, they then work with translating the design into an actual database. This process can also be somewhat difficult for the beginning database student. The normalization process is studied later in the semester, but a few rules are defined for the students to help insure that the resulting database will have as few normalization problems as possible.
The rules for defining the physical database from the E-R diagram are:

- Entity Sets (rectangles) become tables
- Relationships (diamonds) that are one to many, do not become tables, but the key from the one side has to be a field in the many side table
- Relationships (diamonds) that are many to many, become tables and the data in that table are **at least** the keys of the entity sets they relate

2. STUDENT INDIVIDUAL PROJECT

The students then move on to define their own databases project. They are asked to find a real project to implement. The Author approved the project based on the following guidelines:

- ✓ Was the idea a valid one? (a subjective judgment on the part of the Author)
- ✓ Was the problem something the student could reasonably finish in the time allotted?
- ✓ Would the design provide approximately 10 tables once implemented?
- ✓ Were there sufficient relationships among the data? (one to many, many to many, etc…)
- ✓ Were there sufficient attributes to be defined?

Some of the project created were:

- ✓ Database of Fundraisers and their Products
- ✓ Prospective student athlete information for the men's lacrosse team
- ✓ Well device and communication information for an oil & gas company
- ✓ Track services provided by a company with multiple types of audio visual meeting rooms for rent
- ✓ Sales incentive payment information for a financial services company
- ✓ Track information relating to children who attend a Church Sunday School
- ✓ Patient lab test Appointments and results

After approval of the initial database idea, the detailed design was done. Because they were doing work for a real user, it was important that their designs were correct (Choobineh & Lo, 2004). They provided a great deal of information concerning their database project and several passes were made to try to create a correct E-R design. The students presented and defended their designs in class. All students were encouraged to ask questions of fellow students and thus help them to define a correct design. By helping to work out problem with other's designs in the class, essentially the students are designing around 20 to 30 databases (depending on the size of the class each semester). At this point in the semester, the students really did not have enough experience in design work to know if their design was a good one or not. Since the student was creating a "real" project, this Author felt a responsibility to the users who have agreed to participate in the student's learning experience to ensure that the resulting database was correct. Therefore, the author made sure that the designs were correct before allowing the student move on to actually implement the database.

It became apparent to the Author that because each of the students were working on a different project with different issues, a way to organize this information was essential. Therefore, the Author wrote an Access database to store and track all information. The Author also created an Excel spreadsheet with various column headings related to information about the student's individual database project. After the student's database project had been approved, they each filled in a spreadsheet and sent it to the Author. The information was then imported into the Access database. All students were able to "view" the data in the database. Again, a "real world" database was used in the course. And, the "real world" issue of how to integrate spreadsheet data into an existing database was a teaching moment as well. Often data to be stored in a database may come from some external electronic source (Mahoney & Welch, 2006). Figure 3 in the appendix. shows a sample screen of imported data from a student who created a database for a friend who owned a bar. The data on the left of this figure was imported from the student's spreadsheet. And, on the right of this figure you can see several reports that were assigned to this student by the Author.

Students were given access to this database so they could not only track requirements assigned to them but also view a summary of the work of their fellow students. This database also kept track of individual problems or issues that must be addressed by each student. This database was a solution for the Author to the problems of:

- Keeping track of what each student was working on
- Keeping track of problems/ issues that each student must address for his individual database

- Keeping track of custom reports assigned to each student
- Keeping track of individual student E-R diagrams (these were imported into a field that was an OLE object. Please see Figure 4 in the appendix. For a student E-R Example)

Students in the course were also able to see the progress of all other student databases which were real world projects. Thus they experienced many issues beyond their own for their own database. Figure 5 in the appendix shows a sample screen of some issues a student had with his database implementation. The Author entered these issues/problems throughout the semester.

3. SQL LANGUAGE

In the database course, the students also learned SQL (Structured Query Language). They used this with the Author's sample databases as well as their own databases. For the most part, the students did use Access for their projects. However, some used Oracle or Sequel Server. But the students were required to use standard SQL at all times. Even though it is very easy to create tables in Access from the wizard screen they were required to write all data definition language (DDL) statements that would have been necessary for the creation of the tables in their database. These included statements to create the tables, primary keys, foreign keys and various constraints. The students were assigned a number of reports specific to their data by the Author. Again, the Author's Access database was essential to track what was assigned to each student. The students used SQL to create the underlying queries for the reports and then used the software features, such as the access report wizard, to create the actual reports from the SQL query statements. The students were also given a number of assignments using DDL statements and DML (data manipulation language statements). This Author is of the mindset that the actual database product used is not as important as the database topic to be mastered…in this case that being standard SQL.

4. NORMALIZATION

1. Students in a first database course must also be introduced to the process of normalization… (proving that the database design is a good one) Because of the "The rules for defining the physical database from the E-R diagram" referred to earlier in this paper, in most cases, the databases designs did not have too many problems. Students were to prove that all tables were in at least third normal form. A brief definition of first, second and third normal form is provided in Table 1 (http://edn.embarcadero. com/ article /25209)

First Normal Form	**Second Normal Form**	**Third normal Form**
- The table has a primary key. - No single attribute (column) has multiple values. - The non-key attributes (columns) depend on the primary key.	- The tables meet the criteria for first normal form. - If the primary key is a composite of attributes (contains multiple columns), the non key attributes (columns) must depend on the whole key.	- The tables meet the criteria for second normal form. - Each non-key attribute in a row does not depend on the entry in another key column.

Table 1. Brief summary of Normal Forms

5. STUDENT EVALUATION OF EACH OTHER'S PROJECTS

The students started on their individual database project by defending their designs to the entire class. As the semester progressed, they were able to track each other's progress on their database project by viewing the information contained in the course summary database. During the last week of class, they presented their final database projects to the class. They were broken into groups of 4 to 5 students. They presented their database project to their group and they critiqued each other's work. By having a small number of students in each group, it gave them sufficient time to really highlight their work. The Author observed a great deal of pride in their work as each student presented to their groups. They were told at the beginning of the course that they would present their work to the class, so most were well prepared for this event.

6. STUDENT SURVEY AND ATTITUDES

The Author has been teaching database similar to this for a number of years. In the Fall 2013 semester there were 35 students enrolled in the database course. The students reported that they had a high interest in completing the individual database project because it was a "real" assignment for a "real" user. Further, they said that they could not "put down" the database and spent many additional hours getting it to look and run "just right". Many of them also said that they added additional functionality beyond what was required and that they would take an advanced database course if given the opportunity.

In learning to design a database, how effective was it for you to see the critiquing of other student designs?	4.6
How did you feel about seeing other issues/problems students were having in the implementation of their database projects?	3.8
How did you feel about other students seeing your issues and problems with the implementation of your own project?	3.4
How did you feel about seeing the final project of others in the class?	4.9
How did you feel about demonstrating your project to others in the class?	4.2
How did you feel about evaluating other student database projects?	4.5
How did you feel about the amount of time that was spent on design work for the course?	3.9
How did you feel about the various real world designs presented during the semester?	4.8
How did you feel about the course database created to store all student project information?	4.1

Table 2. Student Survey Responses

There is existing research that supports the theory that one course is not enough for the computer student to adequately understand database topics (Schneider, 2006). The students were surveyed about their attitudes toward evaluating fellow student's work. The students were ask to rate each of the following questions on a scale of 1 to 5 with 1 being the least helpful to your learning experience and 5 being the most helpful to your learning experience. The questions and average response are shown in Table 2.

It was the Author's observation that the students are always more interested in a real project. By the end of the semester when the students presented their databases to the class, it was clear to see the pride on the face of those presenting. It was also just as rewarding for the Author to see students giving other fellow students accolades for their work.

Some of the student's comments reported on the survey include:

- "I appreciated the realism in that we'll occasionally be presented with projects where we don't know everything and sometimes have to get more detail"
- "I like the fact that you push designs because they're truly the foundation to a database"
- "It was helpful because I may have had an issue that another student may have been able to help me correct the issues"
- "I feel that it was a good option to demonstrate our databases in front of other students because it may help them with their databases and also they can ask questions on how to implement certain things on their databases"
- "I believe that evaluating other students databases and giving corrective criticism will help them be able to make their databases better"

7. CONCLUSION

This Author was able to bring many personal experiences as a database consultant into the classroom on a daily basis. The students benefitted from the discussions of real encounters with real users and real issues/problems. "Real" projects have always been a great way to get the student involved in what he is learning. This is especially true of a project that the student has selected for himself (Ehie, 2002). Students also learned a great deal from seeing what others are facing in the design and implementation of their projects. Having the students not only work with their own projects but others as well seems to be an aide to the learning process.

Another issue that is of importance and not mentioned in this paper is the area of dishonesty. When using a format such as the one discussed in this paper, there is no possible chance of students copying from each other. In fact, often one student would help a fellow student with a problem he might be having in his database project. This gave the student experiences with issues he might not have encountered in his own database.

If teaching a course in this manner, the Author has a few recommendations:

- ✓ Be careful to limit the scope of each student's project. The student may not realize that the project he wishes to do may be too much for someone new to database work. Also, the project must be completed in the semester time limit. Perhaps deleting some of the functionality he may wish to include will be necessary. By the end of the semester, he should be able to incorporate additional functionality on his own.
- ✓ Make sure students interact with the user often. Any issues concerning the user must be communicated to the instructor as soon as possible. It also might be a good idea for the instructor to briefly interact with the student's user a few times during the semester.
- ✓ A detailed evaluation of the design is necessary in order for the student to complete the database successfully. This will require a significant amount of time on the instructor's part. Because the students are new to database work, and the design is done first in the semester, they will need help to ensure that the design is a solid one.
- ✓ Create some way to organize what each student is working on. This Author wrote a database to store detailed information surrounding the student database projects. This database also allowed the students to keep track of what others in the class were working on. Thus, it gave them another outlet to examine "real world" projects.
- ✓ Find as many ways as possible of bringing real database experiences into the classroom. If the instructor is not doing actual database consulting work, bringing in a database programmer to talk to the class would be advisable. Any way that the students can understand what really happens in the "real world" is advantageous.
- ✓ Make sure the students create sample data for database development. Also, do not include sensitive user information in the student's database prototype. Make sure the user understands that the student's work may be shown in class and therefore user's private information must not be included.
- ✓ Make sure the students understand the grading rubric for the project. This could include an evaluation from the actual user. But is essential that students know how their grade is being calculated. This protects both the student and the faculty member.

This course will help the students become better database designers and implementers. They will come away with a wide range of practical knowledge that is not easily taught in a typical lecture setting. They will succeed in two arenas, technology and communication. It is becoming more and more important to bring "real" experiences into the classroom. The course described here is a step in the right direction (IS2010, Model Curriculum).

8. REFERENCES

Baugh,J.,Davis,G.,Kovacs,P. Scarpino, J., Wood,D. (2009). EMPLOYERS AND EDUCATORS WANT INFORMATION SYSTEMS GRADUATES TO BE ABLE TO COMMUNICATE, Issues in Information Systems, Vol X No 1. pp 198-207, 2009 (Also appears in The Proceedings of IACIS)

Choobineh, J., & Lo, A. W. (2004). CABSYDD: Case-Based System for Database Design. Journal Of Management Information Systems, 21(3), 281-314.

Codd, E.F. (1971)."Further Normalization of the Data Base Relational Model." (Presented at Courant Computer Science Symposia Series 6, "Data Base Systems," New York City, May 24th–25th) IBM Research

Connolly, T., & Begg, C. (2005). Database systems: A practical guide to design, implementation, and management. Reading, MA: Addison Wesley.

Courte, Jill and Bishop-Clark, Cathy (2005). "Bringing Industry and Educators Together," Proceedings of the 6th Conference on Information Technology Education, October 20-22, pp. 175-178.

Date, C. J. (2004). An introduction to database systems. Reading, MA: Addison-Wesley.

Deperlioglu, O., Sarpkaya, Y., & Ergun, E. (2011). Development of a Relational Database for Learning Management Systems. Turkish Online Journal Of Educational Technology - TOJET, 10(4), 107-120.

Ehie, I.C. (2002). "Developing management information systems (MIS) curriculum: perspectives from MIS practitioners", Journal of Education for Business, Vol. 77 No.3, pp.151-7.

Foltz , C. Bryan, O'Hara, Margaret T., Wise, Harold (2004). "Standardizing the MIS course: benefits and pitfalls", Campus-Wide Information Systems, Vol. 21 Iss: 4, pp.163 – 169

Gruba Paul, Al-Mahmood Reem (2004). "Strategies for communication skills development", ACE '04: Proceedings of the sixth conference on Australasian computing education - Volume 30

Hansen, Jorgen (2012). Practical Elements in Danish Engineering Programs, Including the European Project Semester, Industry and Higher Education, v26 n4 p328-335 Aug 2012. 8 pp.

http://edn.embarcadero.com/article/25209

IS2010, Model Curriculum and Guidelines-http--www.acm.org-education-curricula-S%202010%20ACM%20final.pdf

Kim, Youngbeom, Hsu, Jeffrey, and Stern, Mel (2006). "An Update on the IS/TT Skills Gap," Journal of Information Systems Education, Vol. 17, No. 4, Winter 2006, pp. 395-402.

Mahoney, L. S., & Welch, J. K. (2006). Click...and the Database Loads Into Excel. Journal Of Accountancy, 202(3), 53-57.

Philip, George C. (2007).Teaching Database Modeling and Design: Areas of Confusion and Helpful Hints, Journal, v6 p481-497 17 pp.

Report RJ909 (August 31st, 1971). Republished in Randall J. Rustin (ed.), Data Base Systems: Courant Computer Science Symposia Series 6. Prentice-Hall, 1972.

Robbert, Mary Ann; Wang, Ming; Guimaraes, Mario; Myers, Martha (2000). "The Database Course: What Must Be Taught," SIGCSE Bulletin Proceedings of 31st SIGCSE Technical Symposium on Computer Science Education, March, pp. 403-404.

Schneider, B. A. (2006). Hands-on Microsoft Access: a practical guide to improving your Access skills / Bob Schneider. Upper Saddle River, N.J.: Addison-Wesley, 2006.

Seyed-Abbassi, B., King, R., & Wiseman, E. (2007). The Development of a Teaching Strategy for Implementing a Real-World Business Project into Database Courses. Journal Of Information Systems Education, 18(3), 337-343.

Urban Susan D., Dietrich W. Suzanne (2001). "Advanced Database Concepts for Undergraduates: experience with Teaching a Second Course" SIGCSE Technical Symposium on Computer Science Education, February, Volume 33 Issue 1 pp357-361.

APPENDIX

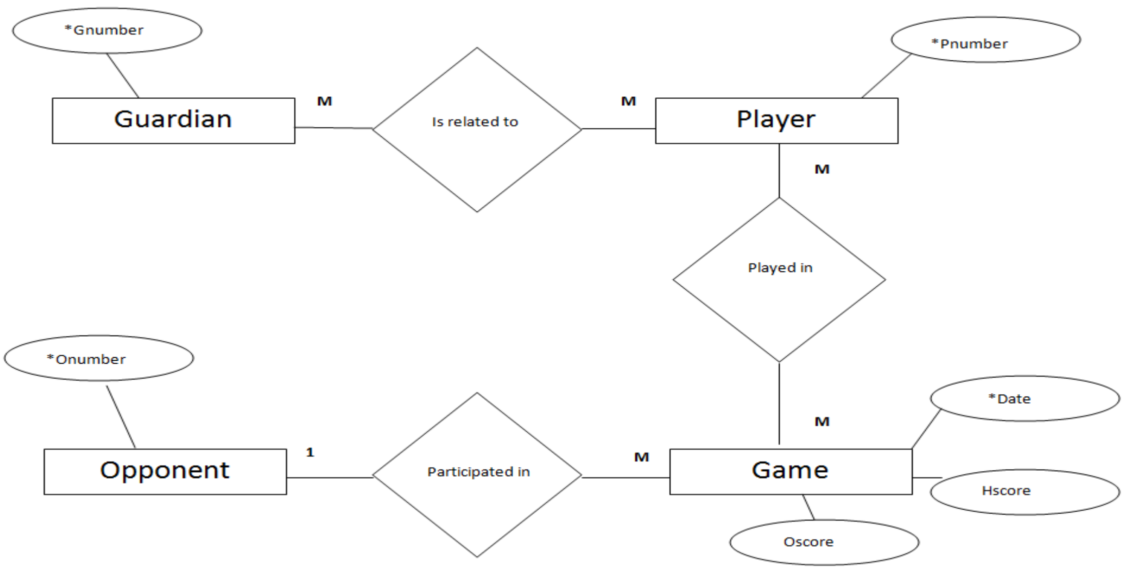

Figure 1. Solution to Youth Soccer Design Problem

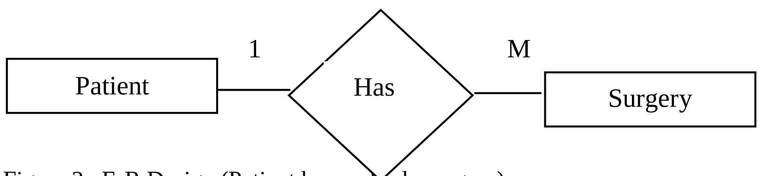

Figure 2. E-R Design (Patient has multiple surgery)

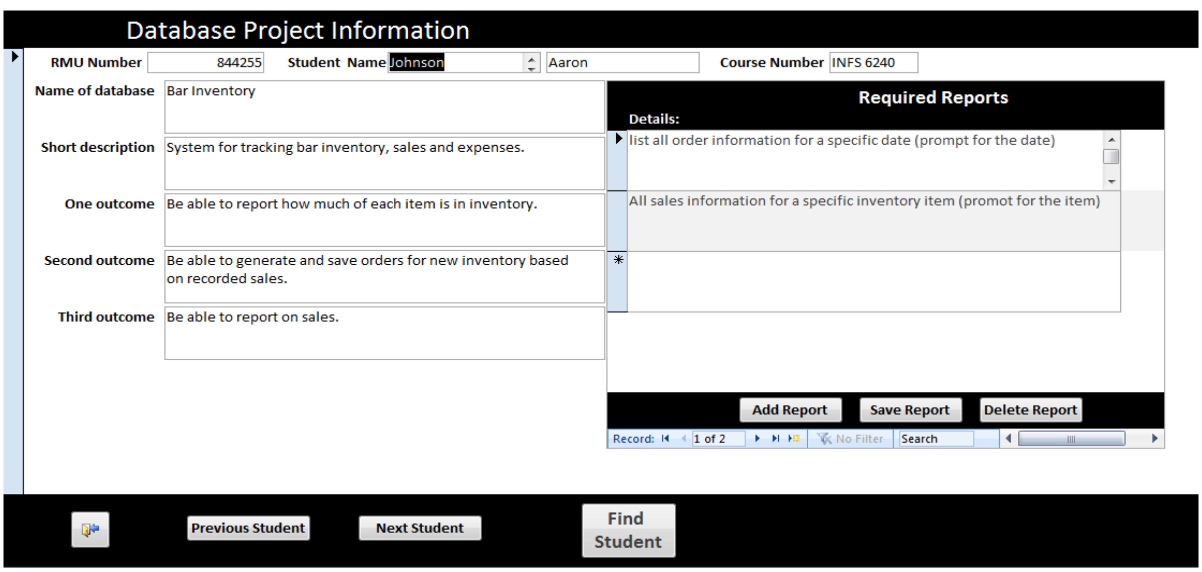

Figure 3. Sample screen of student database project

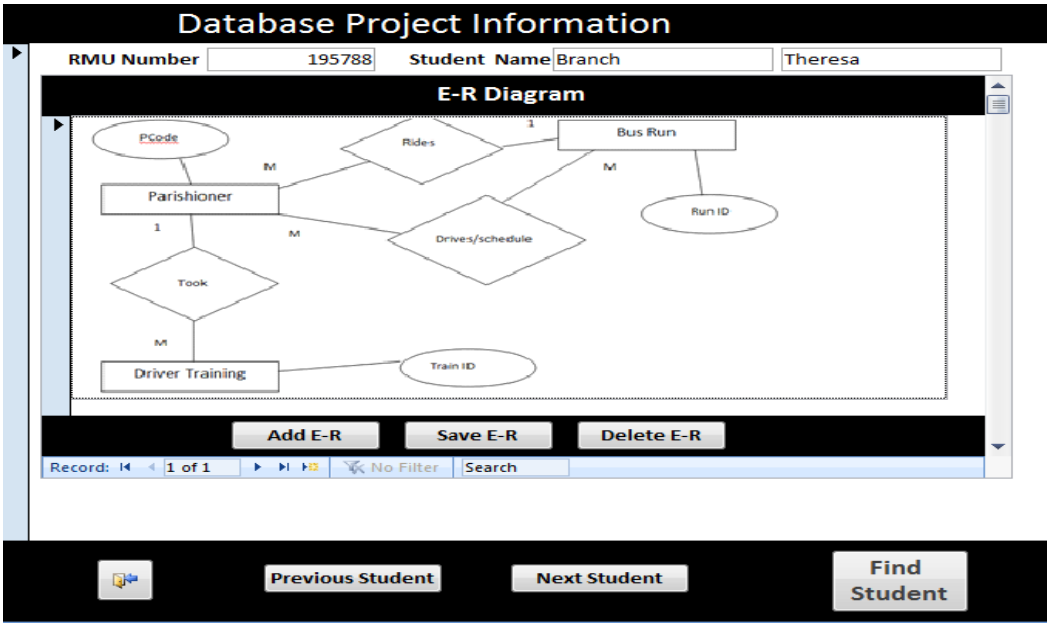

Figure 4. Student stored E-R diagram

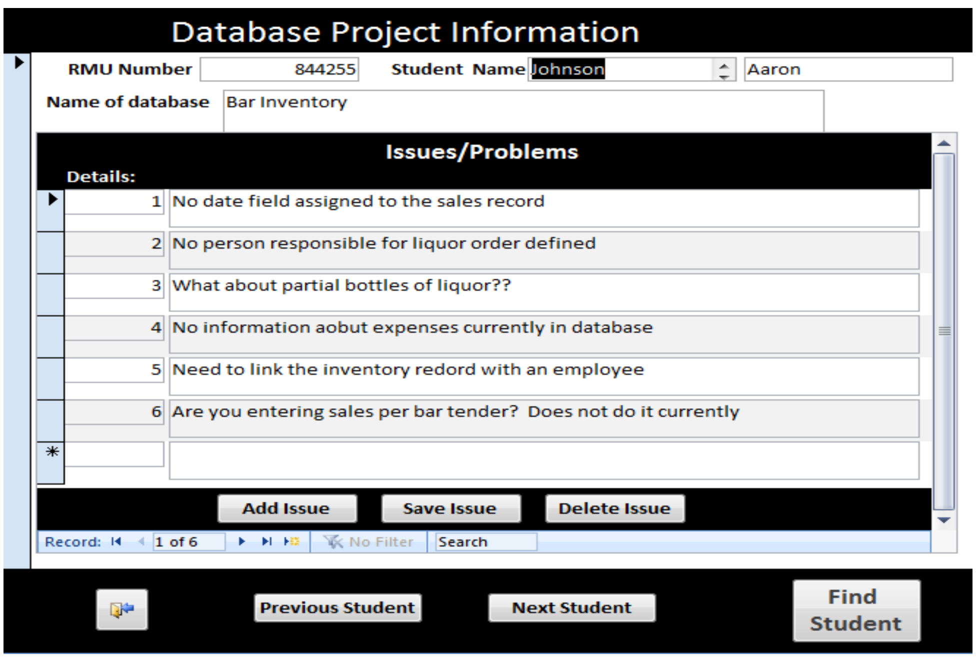

Figure 5. Sample screen of Student database project issues/problems